

**Technical Report
1006**

**Progress Report on the Development
of the Automatic Target Recognition
System for the UGV/RSTA LADAR**

J.G. Verly
D.E. Dudgeon
R.T. Lacoss

21 March 1995

Lincoln Laboratory

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

LEXINGTON, MASSACHUSETTS



Prepared for the Advanced Research Projects Agency under
Air Force Contract F19628-95-C-0002.

Approved for public release; distribution is unlimited.

ADA 294090

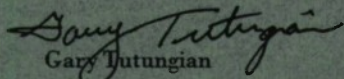
This report is based on studies performed at Lincoln Laboratory, a center for research operated by Massachusetts Institute of Technology. The work was sponsored by the Advanced Research Projects Agency under Air Force Contract F19628-95-C-0002.

This report may be reproduced to satisfy needs of U.S. Government agencies.

The ESC Public Affairs Office has reviewed this report, and it is releasable to the National Technical Information Service, where it will be available to the general public, including foreign nationals.

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER


Gary Tutungian
Administrative Contracting Officer
Contracted Support Management

Non-Lincoln Recipients

PLEASE DO NOT RETURN

Permission is given to destroy this document
when it is no longer needed.

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
LINCOLN LABORATORY

**PROGRESS REPORT ON THE DEVELOPMENT
OF THE AUTOMATIC TARGET RECOGNITION
SYSTEM FOR THE UGV/RSTA LADAR**

*J.G. VERLY
D.E. DUDGEON
R.T. LACOSS
Group 21*

TECHNICAL REPORT 1006

21 MARCH 1995

Approved for public release; distribution is unlimited.

ABSTRACT

This report describes the authors' initial work under the ARPA Unmanned Ground Vehicle (UGV) Demo II Program in the reconnaissance, surveillance, and target acquisition (RSTA) area. The task is to develop the automatic target recognition (ATR) system that will process the imagery from the RSTA laser radar (ladar). A real-time demonstration of this capability is scheduled for the "Demo II" of 1996. All major components of this end-to-end ATR system are discussed, and more details are provided for critical elements that have been built and tested so far. A major topic of interest is the use of sets of azimuth-dependent functional templates and functional template correlation for detecting and recognizing targets in vertically-reprojected forward-looking ladar data. Other topics are the use of height-limited verticality as an interest image for focusing attention and of the Hough transform for getting a preliminary estimate of target orientation. Because the RSTA ladar remains to be procured and built, the report describes also the alternate sources of data that are being used to develop and test the system elements built to date.

TABLE OF CONTENTS

Abstract	iii
List of Illustrations	vii
List of Tables	ix
1. INTRODUCTION	1
2. IMAGE DATA BASES	5
2.1 Introduction	5
2.2 Tri-Service Laser-Radar (TSLR) Data	5
2.3 Hobby Shop (HBS) Laser-Radar Data	11
2.4 Synthetic (SYNTH) Laser-Radar Data	11
3. DESCRIPTION OF PLANNED MODEL-BASED ATR SYSTEM	17
3.1 Introduction	17
3.2 Preprocessing	19
3.3 Detection	23
3.4 Extraction	29
3.5 Recognition	29
3.6 Verification	39
4. EXPERIMENTAL RESULTS	41
4.1 Introduction	41
4.2 Height-Limited Verticality (HLV)	42
4.3 Vertical Reprojection (VR)	42
4.4 Hough Transform (HT) and Peak Detection	59
5. CONCLUSION	79
REFERENCES	81

LIST OF ILLUSTRATIONS

Figure No.		Page
1	Organized data sets of the TSLR data base.	7
2	Azimuth angle convention: case of a tank with a 60° azimuth angle.	10
3	Six TSLR absolute-range images of the M35 truck at 1000 m with resolutions of 0.05 mrad and a frame size of 256 × 128 pixels.	10
4	Six ideal synthetic range images of the M35 truck at 1000 m with pixel sizes of 0.05 mrad and a frame size of 256 × 200 pixels.	15
5	Block diagram of end-to-end XTRS-UGV system.	17
6	Probable source of “ghost” artifact in TSLR data.	21
7	HLV interest images for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad.	43
8	Global overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 1).	45
9	Global overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 2).	47
10	Global overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 3).	49
11	Local overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.	53
12	Local overhead view “count images” for the SYNTH M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.	55
13	Comparison of overhead view “count images” for three real targets and three synthetic targets at nearly identical azimuth angles near 45°. In all cases, target range is 1000 m and reprojection pixel sizes are 0.10 m.	57
14	HTs of the local “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.	61
15	HTs of the local “count images” for the SYNTH M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.	63
16	HTs angle convention: case of a tank with a 30° azimuth angle.	65
17	Relation between azimuth angle and HTs angle for a variety of situations.	65

LIST OF ILLUSTRATIONS

(Continued)

Figure No.		Page
18	Raw range images, overhead view "count images", and HTs for SYNTH (top row) and TSLR (bottom row) M35 truck.	67
19	Raw range images, overhead view "count images", and HTs for SYNTH (top row) and TSLR (bottom row) M35 truck.	69
20	Signed error of HT azimuth estimate for the TSLR M113A2 APC, M35 truck, and M60A1 tank.	73
21	Signed error of HT azimuth estimate for the SYNTH M113 APC, M35 truck, and M60 tank.	75
22	Signed error of 1-D HT azimuth estimate for the SYNTH M113 APC, M35 truck, and M60 tank.	77

LIST OF TABLES

Table No.		Page
1	Estimated Angular Pixel Sizes (mrad) for TSLR Data	8
2	Angular Pixel Sizes (mrad) for HBS Data	11

1. INTRODUCTION

This report is a slightly revised and expanded version of an informal semiannual report submitted to ARPA on April 15, 1994 and titled "UGV/RSTA Progress Report." The semiannual covered the period from October 1, 1993 through March 31, 1994 and described the authors' efforts under the ARPA Unmanned Ground Vehicle (UGV) Demo II Program in the reconnaissance, surveillance, and target acquisition (RSTA) area [1]. The current report presents the state of the authors' work as of the "Demo B" milestone of June 1994. A condensed version of this report will be found in the proceedings of the 1994 Image Understanding Workshop. The present report contains more pictorial illustrations of the experimental results and uses color to enhance the presentation of these results.

The overall goal is to develop a model-based target recognition system that will run in real-time on the Semiautonomous Surrogate Vehicle (SSV) [2] as part of Demo II of June 1996. The recognition system, called XTRS-UGV (eXperimental Target Recognition System for the UGV), will use imagery from the RSTA laser radar (ladar) sensor to identify targets in the Demo II target set. Over and above the primary goal, if the work proceeds on or ahead of schedule, the authors hope to demonstrate the fusion capability of XTRS-UGV as a laboratory demo in 1996. It is also conceivable that the system could be used to demonstrate ladar/forwardlooking infrared (FLIR)/video fusion as part of the main demo in 1996.

The proposed techniques, which include height-limited verticality (HLV) interest images, re-projection, functional-template correlation (FTC), and back-end model matching, will provide the SSV with a real-time target recognition capability for the Demo II scenario and target set. These claims are based in part on the authors' success in other ladar and synthetic aperture radar (SAR) target recognition programs, as discussed in their publications (e.g., in [3, 4, 5]). HLV is a detection technique that was developed to find vertically extended objects with predefined height limits in ladar range imagery. It is used with other detection techniques to form interest images. Interest images facilitate the integration of information for focus-of-attention processing (i.e., detection within the ladar image). The plan is to use them for combining ladar range and intensity information initially, and it is pointed out that it is feasible to incorporate FLIR information into the same process. FTC applied to reprojected ladar range imagery should provide accurate pose estimation and initial hypothesis formation with modest computational requirements. Furthermore, reprojection can be applied to other sensor modalities (e.g., ladar intensity and FLIR intensity), and the reprojected data can then be processed by appropriate functional templates (FT). Applying FTs to a number of interest images derived from one or more imaging modalities is a powerful, proven, general-purpose mechanism for simultaneously performing fusion and recognition directly at the pixel level. Back-end model matching against image-derived features provides a robust method of verifying target hypotheses. It is felt that these ideas (HLV, interest images, reprojection, FTC, and back-end model matching), as planned for the UGV Demo II Program, represent an advance in the state of the art.

Because the primary goal is to recognize targets, progress will be measured by using standard metrics such as confusion matrices and plots of percentage of correct recognition vs percentage of false recognition. Of course, the results will depend heavily on the degree of difficulty of the recognition problem, e.g., sensor characteristics (such as resolution), clutter state, number of pixels on target, intertarget similarity, tilt, state of articulation, obscuration, extinction coefficient or, equivalently, carrier-to-noise ratio at the receiver. Assessing performance as a function of these independent variables will be more difficult than simply assessing performance on a fixed, given data set. Furthermore, there are a number of internal functions the performance of which will also be assessed to some degree (e.g., focus of attention, pose estimation).

Development risk is minimized by adopting the following strategies. First, the authors plan to use and/or adapt, to the greatest extent possible, techniques that have proven successful in other ATR systems built by the authors, not only for forwardlooking and downlooking laser-radar imagery, but also for SAR imagery [3, 4, 5]. Second, initial efforts will concentrate on the implementation and testing of the algorithms that are believed to be the most important in the overall system. Third, the goal is to achieve recognition capability early-on, possibly skipping temporarily some processing steps that are not in the critical processing path. (Note that some form of recognition will already be achieved without having to implement the model-based matching in the back-end verification stage.) Finally, the end-to-end system, as well as the individual system components, will be continuously tested on large, well-organized data sets.

In the author's view, the main challenges for XTRS-UGV will be the relatively small number of pixels on target (because of the coarse angular resolution of 0.3 mrad of the planned RSTA ladar), the potential articulations of some targets, the need for estimating the ground orientation, and the occlusion effects. The effects of motion also constitute a major difficulty, but the current working assumption is that XTRS-UGV will be provided with motion-compensated imagery.

Integration of the authors' target recognition system into the SSV processing system should be straightforward. Because of the small field of view of the ladar, it is necessary to cue it with detections from the FLIR or video sensors. After the ladar is pointed and an image is produced, the system can be used to find and recognize the target(s) within the ladar field of view. It may be possible to improve recognition performance somewhat by using map data and pointing information to help estimate the ground position and orientation and to eliminate certain areas of the image from further consideration.

The model-based approach that is being proposed also provides mechanisms to incorporate information from other sensors. Interest images have proven to be a robust method of fusing measurements from several sensors. Likewise, the models that will be used for functional-template-based recognition (and back-end model-based verification) could be extended to incorporate FLIR and video features in addition to the ladar range and intensity features. Algorithms from other contractors for tasks such as focus of attention and feature extraction could be integrated into XTRS-UGV.

The current plans are to develop a baseline recognition system using the available data to establish initial performance results. The baseline system was shown as a technology demonstration at the Demo B meeting in June 1994. The next status report will be presented at the Image Understanding Workshop in November 1994. After establishing a baseline system and benchmarking its performance, the authors shall refine those parts of the system that they feel will result in better overall performance. After refinement of the XTRS-UGV system, they will work closely with Martin-Marietta Denver to transfer the baseline recognition system to the SSV processing system. The plan is to show a laboratory demonstration using the SSV hardware as part of Demo C in June 1995. After that, the plans call for refining the system to achieve adequate performance for the Demo II scenario in an appropriate processing timeline for the overall UGV system. This will be demonstrated as part of Demo II in June 1996. If the authors meet or exceed their expectations, they plan to develop the capability to use other sensor information (FLIR, video) in concert with the ladar.

Section 2 of this report discusses ladar data that is being used to develop the recognition system initially, in the absence of data from the real sensor. In Section 3, the planned XTRS-UGV system is described in detail. Not all components of the algorithm are implemented at this time because the approach taken is to work on higher-risk components first. Section 4 presents experimental results obtained from the implemented system components, using the available ladar data. Section 5 contains the conclusion.

2. IMAGE DATA BASES

2.1 Introduction

This section describes the images that have been acquired or created so far for developing and testing the XTRS-UGV.

The three current sources of imagery are the following: (a) data collected with the Tri-Service Laser-Radar (TSLR) sensor, (b) data collected during the Hobby Shop (HBS) collection exercise, (c) synthetic data (SYNTH) generated using a synthetic generator developed in-house. Each of these types of data is described below.

2.2 Tri-Service Laser-Radar (TSLR) Data

The TSLR sensor essentially consists of a CO₂ STARTLE ladar with an added passive channel.

Several thousand TSLR image sets were obtained from the Night Vision Lab through Richard Peters. Each set contains five images, each corresponding to one of the following imaging modalities: (1) fine, absolute range, (2) fine, ambiguous range, (3) ladar intensity, (4) ladar Doppler, and (5) FLIR [6]. All the images of interest were collected at Fort A.P. Hill, VA (Drop Zone) in the October to December 1989 time period. In all the experiments described noted in this report, only the fine, absolute range imagery is used.

2.2.1 Data Base Management System

The TSLR data made available to the authors is not organized in a way that is convenient for the development and testing of ATR systems. For example, there is no easy way to extract a set of logically related views, such as all the views of a tank systematically rotated at increasing azimuth angles. The header files (one per image) must be examined to extract the necessary information. To deal with the large amount of data provided and to construct the image sequences that were needed, one was forced to develop a simple UNIX-based data base management system (DBMS) that relies on the header files and is robust to typos and errors in the filenames and header fields.

In the first step, the DBMS reads the thousands of header files that accompany the image files and extracts (sometimes with considerable difficulty, because of typos and other data entry errors) a set of preselected fields, such as the target class (e.g., APC, TANK, TRUCK), the target type (e.g., M113A2, M35, M60A1), the range to the target, the angular resolution, the frame size (e.g., 256×128), the azimuth angle, the view index (many targets were imaged several times in each of their poses, presumably to look at random effects in the imaging process), the flag indicating whether calibration plates were present, etc. As a result, each "quintuplet" of images is characterized by a single entry in the data base, and each such entry is characterized by a number of attribute/value pairs.

In the second step, the data base so created are examined with a simple query language. A typical use of this language is as follows: “Create a sequence (at increasing azimuth angles) of fine, absolute range images corresponding to the first view (among multiple looks) of the M35 truck at 1000 m with a nominal angular resolution of 0.5 mrad, a frame size of 256×128 pixels, and with no calibration plates present.”

The output of the query language for such a request is a file containing a list of the filenames of the files containing the desired views. The azimuth angle is also recorded for future use. At this point, the desired files are extracted from tape or disk and converted from the original ARF format to the VIFF format of Khoros [7]. Each sequence of interest must be examined manually to correct a number of errors due to incorrect data entry (either in the headers or in the filenames). A typical error is an incorrectly recorded angular resolution. This error is easily spotted when the image sequence is played as a movie. It is more difficult to find out whether a missing image actually exists under an incorrect filename or with an incorrect header. In addition, any frame that is heavily corrupted by abnormal artifacts is discarded from the sequence.

2.2.2 Development and Test Data Base

The final outcome of the process just described is a series of well organized data sets that can be used to develop and test the various elements of the proposed ATR system. The 12 data sets that are used extensively are summarized in Figure 1.

As shown in the figure, image sequences (of each modality) were constructed for the M113A2 APC, M35 truck, M2 Bradley fighting vehicle, M551 tank, M60A1 tank, and M60A2 tank at each of the available angular resolutions. The most detailed images correspond to a resolution of 0.05 mrad (with a beam footprint of 5 cm at 1 km) and the less detailed images to a resolution of 0.1 mrad (with a beam footprint of 10 cm at 1 km).

TSLR data is generally available at three angular resolutions referred to as “A” (0.2 mrad), “B” (0.1 mrad), and “C” (0.05 mrad). However, the available data does not include any images at the coarsest angular resolution of 0.2 mrad. This is unfortunate because this resolution would be closer to the planned resolution of 0.3 mrad for the UGV/RSTA sensor. It is worth emphasizing that the best resolution of 0.05 mrad is a factor six better than the planned UGV resolution of 0.3 mrad. This should be kept in mind in evaluating the experimental results presented in later sections.

From Figure 1, one can see, for example, that each sequence for the M35 truck consists of 70 images (each image in a sequence corresponding to a distinct azimuth angle). For each target, the nominal separation between successive azimuth angles is 5° (in which case a complete sequence would consist of 72 individual images). However, for some targets, there are a number of images that are missing or unusable. (The best sequences have 70 images and are thus missing only two images.) Note that the true separation between successive views is not always 5° or a multiple thereof. This issue is further discussed below in the section on “ground truthing.”

ANGULAR RESOLUTION (mrad)	NBR OF TARGET IMAGES (1000 m, ~5° Azimuth Increments)						
	APC	TRUCK	TANK				ALL
	M113A2	M35	M2BRAD	M551	M60A1	M60A2	
0.10 ("B")	61	70	64	24	22	22	263
0.05 ("C")	64	70	59	18	52	22	285
ALL	125	140	123	42	74	44	548

IMAGE TYPES:**FINE, ABSOLUTE RANGE (4 Bytes)****FINE, AMBIGUOUS RANGE (2 Bytes)****INTENSITY, PASSIVE, DOPPLER (2 Bytes)****IMAGE SIZES: 256 × 128 PIXELS****FIELD OF VIEW:****"B" = 1.16 × 0.52 deg****"C" = 0.58 × 0.26 deg***Figure 1. Organized data sets of the TSLR data base.*

Viewing each sequence as a movie (e.g., through the Khoros program "animate") is extremely useful for exploring the data. In addition, each image in a sequence can be processed through any sequence of algorithms and the resulting set of images can also be viewed as a movie.

Figure 1 indicates that the 12 canonical sequences consist of a total of 548 distinct image "quintuplets" from the available TSLR data base.

The list of filenames that make up each of the TSLR sequences could be made available to any UGV/RSTA participant who has a need for them.

2.2.3 Angular Resolution and Angular Pixel Sizes

Even though angular resolution is an important system parameter, knowing the exact angular intervals at which the field of view is sampled (and therefore the resulting angular sizes of each pixel) is far more important for correctly applying a number of geometric transformations to the range imagery (and/or to any image that is pixel-registered with the range imagery). A prime example of such a transformation is the vertical overhead reprojection described in Section 3.

Because the horizontal (x axis) and vertical (y axis) pixel sizes are important parameters (and apparently not readily available), their values were determined through careful examination of the

data and blueprints of the targets. The general procedure is as follows. First, one tries to find range images that show views of objects that are as close as possible to “broadside.” Second, one takes various horizontal and vertical measurements of clearly visible features (the measurements are in units of pixels). Finally, these measurements in pixels, together with the corresponding measurements in meters (from blueprints) and the range to the objects, are used to determine the angular pixel sizes.

Because the azimuth information provided in the image headers cannot always be relied upon to identify broadside views of objects, the following iterative procedure was used to find such views. First, the image of a particular sequence that appeared to be closest to a broadside view was displayed, measurement were taken (in units of pixels) and related to blueprint measurements (in meters). This led to first estimates of the the x and y sizes of a pixel (this was done at the highest “C” resolution). These values were then used in conjunction with a reprojection algorithm (described in Section 3) to produce an overhead view of the target present in the imagery. With a sequence of such views around broadside, one could determine which range image was actually closer to broadside. If a new image was selected as being closer to broadside, the whole process was repeated to arrive at better estimates of the pixel sizes.

This estimation process was carried out for the M113A2, M35, and M60A1, and the best estimates are shown in Table 1.

TABLE 1
Estimated Angular Pixel Sizes (mrad) for TSLR Data

	M113A2	M35	M60A1	MIT LL	ALLIANT
X (horiz)	0.0413	0.0395	0.0384	0.0397	0.0395
Y (vertic)	0.0347	[0.0403]	0.0369	0.0358	0.0310

The first three columns are self-explanatory, the fourth corresponds to averages across all three objects (excluding the vertical estimate for the M35 that appears to be an outlier), and the fifth provides the corresponding values given by Alliant Techsystems in a report made available to us recently [8]. Because the Alliant Techsystems estimates were apparently obtained from the lower resolution (“B”) data, the authors decided to adopt their own average estimates in all the experiments of Section 4.

2.2.4 Establishing the Ground Truth

The azimuth angles given by the TSLR data headers are often unreliable. (Some azimuths appear to be off by as much as 10°.) In order to correctly score a variety of algorithms (preliminary

pose orientation by Hough transform (HT), refined orientation by FT, etc.), it was found necessary to “ground truth” the TSLR imagery. (A more correct term would be “image truth” because the truth is found from the imagery itself.) Ideally, ground truthing would consist of establishing the complete pose of the object (position and orientation). However, for immediate purpose, it was sufficient to only establish the true target azimuth. Just as the reprojection algorithm (to be discussed in detail later) was found useful to find broadside views of targets (to estimate the pixel x and y sizes), the very same algorithm was found useful to estimate target orientation for each image in a sequence.

The procedure consisted of first reprojecting each image (using the correct, previously determined pixel sizes) and then rotating the resulting overhead-view image by various angular amounts until the reprojected object could be brought back into a standard position (front view). The Khoros “cantata” framework was extremely well suited for performing this angle estimation procedure interactively and quickly.

So far, the true target orientation was found for each image in the sequences for the M113A2 APC, M35 truck, and M60A1 tank, at the highest “C” resolution (a total of 186 images). These estimates (and others to follow) could be made available to any UGV/RSTA participant who has a need for them.

For future reference, it is useful to state the adopted convention for azimuth angle measurements. Figure 2 shows that an object with its front facing the sensor has, by definition, an azimuth of 0° . As the object rotates clockwise (as seen from above), the azimuth angle increases. Thus, 90° corresponding to a side view (broadside) with the target front facing left (as seen in the image). This convention is also that of the TSLR data and is believed to be fairly common in the ATR field. It is also used for the other types of images described below.

2.2.5 Data Set Names

For ease of reference, a systematic naming convention was adopted for the various data sets (12 of which are shown in Figure 1). For example, the sequence of 61 images for the M113A2 APC at 1000 m and 0.1-mrad resolution is named M113A2-1000-B6, where “B” is the standard TSLR symbol for 0.1 mrad angular resolution and “6” is the standard TSLR symbol for a frame size of 256×128 pixels. Thus “B6” corresponds to a field of view of $3.2 \times 1.6^\circ$. The data sets M113A2-1000-C6 (64 images), M35-1000-C6 (70 images), and M60A1-1000-C6 (52 images) are used extensively in the experimental results presented in Section 4.

2.2.6 Example Imagery

Figure 3 shows six of the 548 fine-range images in the TSLR data base. These images corresponds to six views of the M35 truck at approximately 30° increments beginning with the front view at 0° , for a range of 1000 m, a resolution of 0.05 mrad, and a frame size of 256×128 pixels. (These views are part of the M35-1000-C6 data set.)

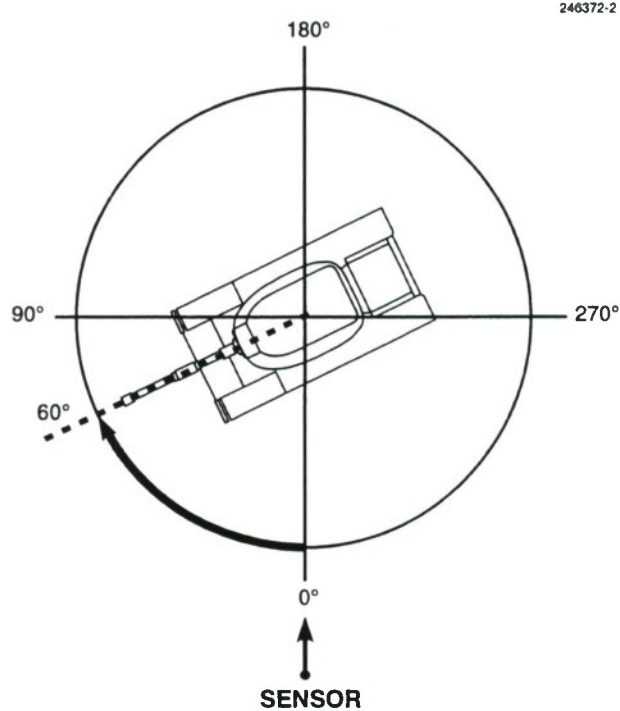


Figure 2. Azimuth angle convention: case of a tank with a 60° azimuth angle.

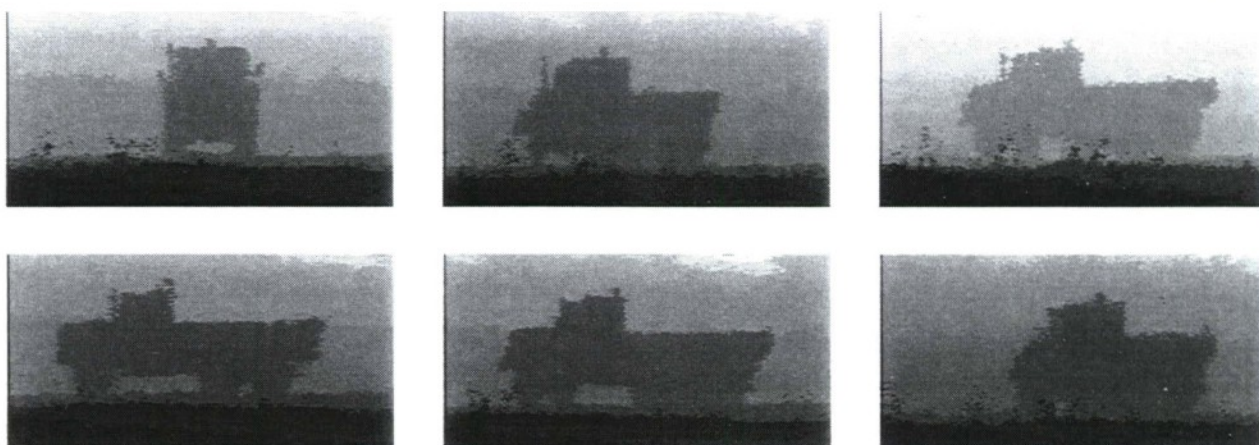


Figure 3. Six TSLR absolute-range images of the M35 truck at 1000 m with resolutions of 0.05 mrad and a frame size of 256 x 128 pixels. The respective ground-truth azimuths are 0° , 34° , and 60° for the top row, and 89° , 117° , and 151° for the bottom row.

2.3 Hobby Shop (HBS) Laser-Radar Data

The HBS data was collected at Fort Carson, CO, in November 1993 [9]. Imagery was collected using three sensors: an FLIR (from AMBER), a ladar (from Alliant Techsystems), and a color CCD camera. The authors have downloaded all the relevant HBS data shortly after it became available on the Martin-Marietta Denver ftp server (MMC). They have also acquired the corresponding photo-CDs.

Most of the images of interest have been examined, but the algorithms developed so far have not applied to any of the images. However, the plan is to exercise all of the existing algorithms on these new images. Except for the initial conversion to a compatible unit of range, all of the algorithms are believed to have already built-in all the parameters that will be necessary for dealing with the new imaging conditions.

For future reference, note that the x and y pixel sizes that were provided to the authors, at their request, by Ross Beveridge are as given in Table 2.

TABLE 2
Angular Pixel Sizes (mrad) for HBS Data

	Predicted	Measured
X (horiz)	2.2	2.28
Y (vertic)	2.639	2.61

Note that Ross Beveridge refers to these numbers as resolutions. However, his method of calculation seems to indicate that these values are indeed pixel sizes. Because of the uncertainty, the suggested interpretation of the numbers in the table should be questioned before the numbers are actually used.

2.3.1 Example Imagery

The reader is referred to the recent report describing the HBS collection exercise for examples of HBS imagery [9].

2.4 Synthetic (SYNTH) Laser-Radar Data

In their prior work with ladar imagery, the authors had developed a flexible software tool for generating synthetic ladar images (range and intensity) [3]. This tool was quickly upgraded to serve the UGV/RSTA program goals. An important addition was the capability of producing

synthetic images in the native VIFF format of Khoros. Hence, early in the first six months of work, it became possible to test the various algorithms both on real data and on synthetic data.

The reason for looking at synthetic data is twofold: first to augment the TSLR and HBS data bases; second to help create target models (initially in the form of FTs).

2.4.1 Synthetic Data Generator Capabilities

The authors' in-house synthetic data generator can use three different types of CAD models. However, the current UGV/RSTA needs are well served by the 17 models developed by ERIM for the PAIRSTECH program. These models are of modest complexity (say, at most 200 facets) and thus imagery can be generated quickly. The models that are being used are not the original PAIRSTECH models, but modified versions where, among other things, the bottoms of the models were closed off with plates and the facets of the models grouped in parts, in such a way that parts can be rotated whenever appropriate. For example, in the case of tanks, the turret can be turned left or right and, at the same time, the gun can be moved up and down. Furthermore, one can attach optional antennas to most of the models.

Realistic noise can be added to both the intensity image and the range image. (These "noises" are interrelated because a low intensity return typically leads to a missing value or outlier.) However, there are cases where ideal, purely geometric range images are also useful (typically for building FT models). All the synthetic range images used in the experiments described later are noiseless.

Scenes can be constructed that consists of several objects (possibly obscuring each other) on a ground plane with movable background walls at arbitrary orientations. Obscuration can easily be simulated, e.g., by creating telephone poles or walls that partially obscure targets.

The models available for synthetic data generation are the following: BMP, BDRM2, BTR60, M109, M113, M151, M1967, M1, M35, M36, M48, M60, T55, T62, TAB72, UAZ469, ZIL130. In addition, a civilian FORD VAN was constructed from scratch.

The synthetic data generator is built in such a way that all CAD models are converted to a common, internal, facet-based representation that is then used for scene generation. In addition to the capability of converting the PAIRSTECH models, one can also convert very detailed models (having about 30,000 facets) obtained a number of years ago from the Air Force Armament Laboratory at Eglin AFB.

Even though the current synthetic-image-generation capabilities serve current purposes well, the plan is to examine the capabilities of the SAIL and to consider the use of BRL models.

2.4.2 Development and Test Data Base

The authors have generated 72-image data sets (with exact 5° azimuth increments and no azimuth coverage gaps) corresponding to the TSLR data sets M113A2-1000-C6, M35-1000-C6, and M60A1-1000-C6 (discussed above). Because the primary goal in using synthetic data is to guide

the design of FTs, the synthetic-image sequences generated to date correspond to noiseless, ideal, purely geometric range images with the object floating in space (i.e., with no ground and no other backgrounds of any kind). In addition, the range values in these range images are as exact as they can be, i.e., they are not quantized to the typical range quantization interval of 1 ft. Noisy, quantized range and intensity images could be generated and they will be generated as the need arises, most probably to augment the test data base.

Finally, an obvious, important advantage of synthetic data is that the ground truth is perfectly known and, therefore, does not need to be established experimentally.

2.4.3 Data Set Names

The names of the primary synthetic data sets are M113-1000-005-T, M35-1000-005-T, and M60-1000-005-T, where 1000 is the range in meters, 005 a symbol for the angular resolution of 0.05 mrad and "T" indicates "target-only," i.e., with no backgrounds of any kind. These data set names will appear frequently in discussions of the experimental results.

It should be clear that, even though most of the results reported below correspond to the M113, M35, and M60, similar results could easily be generated for the other 14 models in the CAD model library.

2.4.4 Example Imagery

Figure 4 shows 6 of the 216 (72×3) ideal, noiseless, unquantized range images currently in the SYNTH data base. These images correspond to six views of the M35 truck at exact 30° increments beginning with the front view at 0° , for a range of 1000 m, a resolution of 0.05 mrad, and a frame size of 256×200 pixels (these views are part of the M35-1000-005-T data set). Note that the truck is "floating in space." (The reasons for this will be discussed in Section 3.)

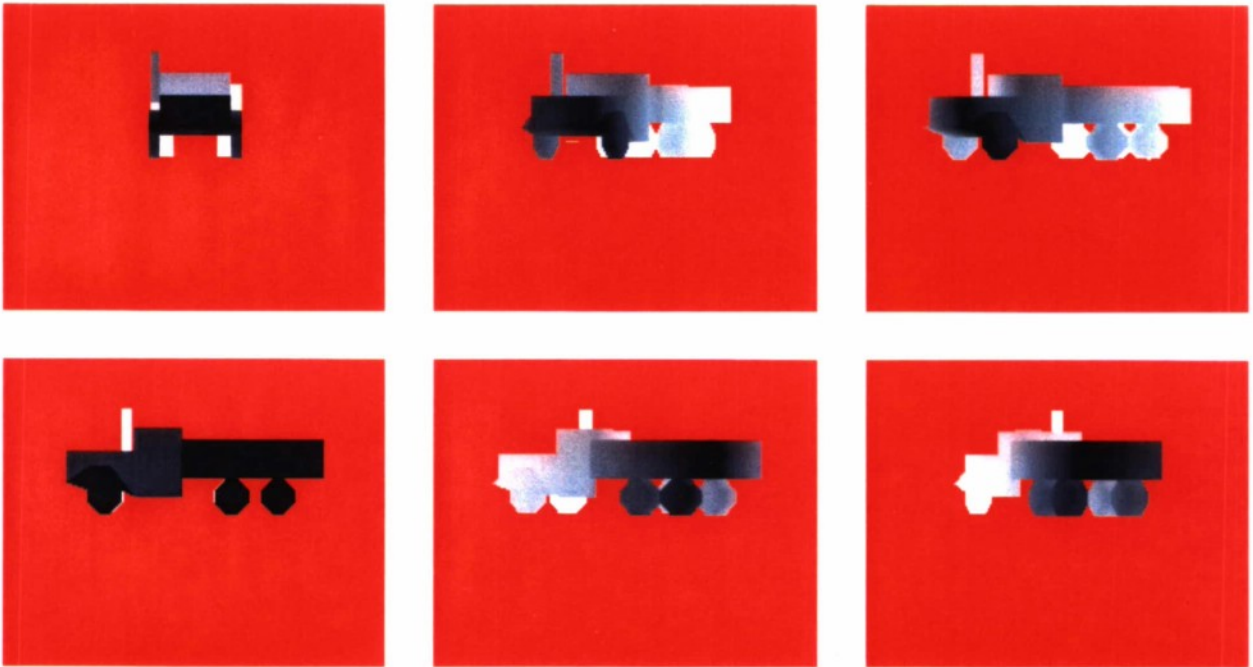


Figure 4. Six ideal synthetic range images of the M35 truck at 1000 m with pixel sizes of 0.05 mrad and a frame size of 256×200 pixels. The respective ground-truth azimuths are 0° , 30° , and 60° for the top row, and 90° , 120° , and 150° for the bottom row.

3. DESCRIPTION OF PLANNED MODEL-BASED ATR SYSTEM

3.1 Introduction

Figure 5 shows a high-level block diagram of the UGV/RSTA model-based ATR system that is being developed. The working assumption is that systems developed by others will dictate where to point the ladar sensor and when to record the ladar images. Therefore, one assumes that the input to the system of Figure 5 is a pair of pixel-registered range and intensity ladar images. (So far, the intensity imagery, which was not provided as part of the HBS data, has been ignored, but the authors' prior work with ladar imagery has led them to develop effective methods for exploiting the intensity image jointly with the range image.) Even though advanced detection and recognition methods could be developed for dealing with sequences of ladar images (range and intensity), this is not part of the current plan.

In spite of the current focus on ladar imagery, many of the ideas that are being developed could be extended to encompass other imaging modalities such as FLIR. Whenever appropriate, one will indicate which parts of the system could be applied to these other types of images

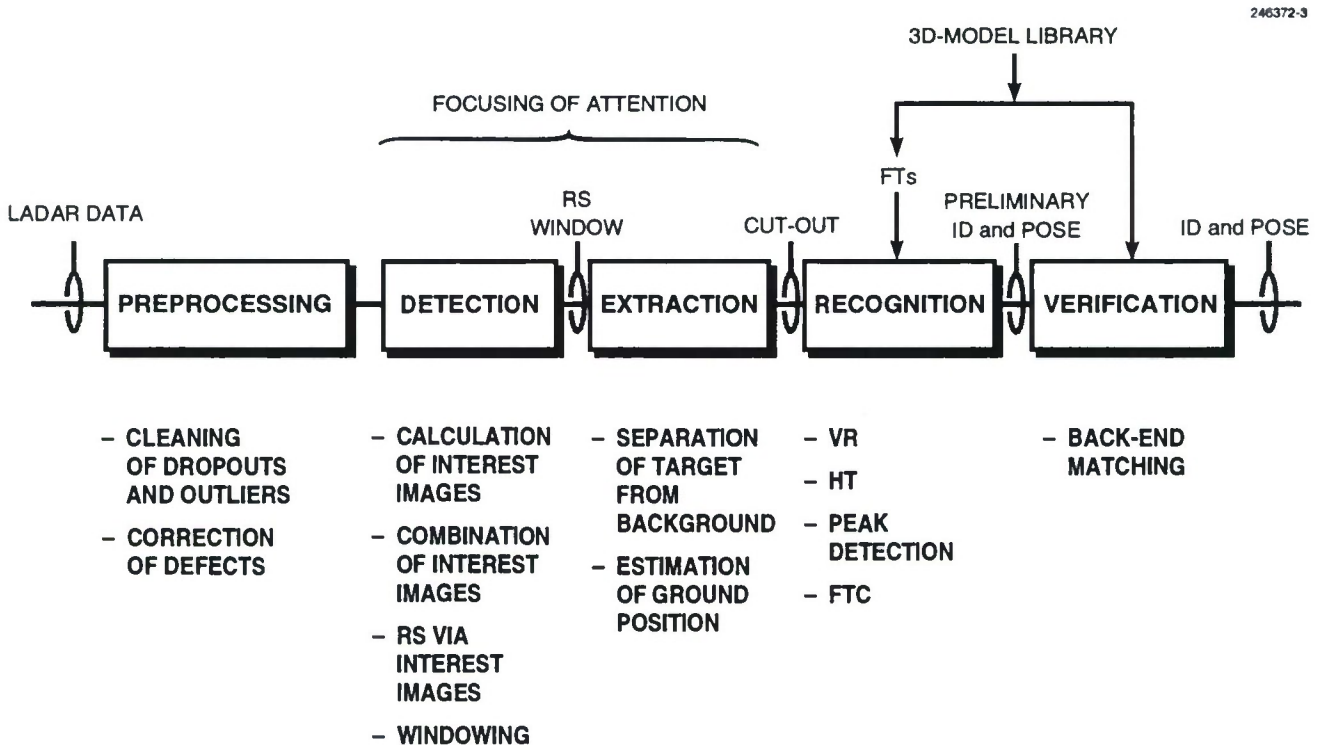


Figure 5. Block diagram of end-to-end XTRS-UGV system.

A brief overview of each of the elements of the block diagram of Figure 5 is now presented.

1. **Preprocessing** may include a number of steps intended to correct potential defects of the input imagery. One such step might be to clean out “missing values” (i.e., dropouts and outliers) from the range imagery.
2. **Detection** should not be confused with the main detection system developed by others to locate candidate targets in wide-field-of-view imagery (most probably FLIR). However, once this main detection system has detected a candidate target and aimed the ladar in its general direction, the ladar imagery that is passed on to the present system might still contain multiple targets, each potentially located anywhere in the image frame. Hence the detection step described in this report is tasked with focusing the system’s attention to “interesting” areas of the ladar imagery. In fact, this second detection step gives an opportunity to apply powerful detection methods that are specific to range imagery and complement other detection methods based on FLIR imagery only. Some of the methods that are being developed could in fact be used in the initial detection step (currently developed by others) if wide-field-of-view ladar data were available. The typical output of the current detection step is one or more windows (with range-dependent sizes) that are likely to be centered on “interesting” areas of the images (both in the plane of the image and in range).
3. **Extraction** is applied to each individual detection window. The goal here is to extract the object that most probably triggered the generation of the window that is currently being investigated. The methods that will be used in this step are typically quite different from the ones that are used in the detection step. Ideally, extraction will act as a cookie cutter that will isolate the object of interest from all other things (ground, background, trees, etc) in the window of interest. Note that this cookie-cutter operation can preserve all of the information within the shape that is cut out, i.e., range, intensity, FLIR (if applicable), etc.
4. **Recognition** is the process that will decide whether the shape that has been cut out is one of the targets in the library or a false alarm (e.g., created by a rock or some other target-like object). If the “cut out” matches targets in the model library, the recognition step will provide the identity of the best match, a corresponding score or degree of confidence, and the estimated pose (i.e., position and orientation) of the target. The current plan is to rely on FTs to implement recognition.
5. **Verification** does not attempt to modify the identity found in the previous step. It simply tries to verify that the target hypothesis proposed in the previous step is indeed valid. The verification step can use more complicated matching methods because it does not have to consider all the targets in the model library. Note that it would be simple to generalize the system by allowing “verification” to choose the best among several hypotheses produced by “recognition.”

Each of the five elements of the block diagram are now discussed in detail. In each case, one gives an overview of the processing element, the principles involved, and a status report. The strategy has been to quickly implement the processing modules that are regarded as critical. Therefore, some of the processing modules that are described have not yet been implemented. More details will be provided for the steps that have actually been implemented and tested. The corresponding experimental results are discussed separately in Section 4.

3.2 Preprocessing

Preprocessing is intended to prepare the range (and registered intensity) for subsequent processing. At the very least, preprocessing should ensure that the range units are correctly interpreted (e.g., these units are different for the TSLR, HBS, and SYNTH data). More importantly, preprocessing will correct any geometric distortion that might affect the imagery (e.g., distortion due to motion) and will clean out any noise and any other artifacts from the imagery.

Noise in range imagery is of a very particular nature and is highly dependent on the sensor being used. Range values at some pixels may be missing or completely erroneous. The term “dropout” is often associated with pixels for which no lidar returns were recorded (either because the beam did not encounter any reflective surface, as is the case for the sky, or because the beam was deflected away from the receiver). The term “outlier” is often used to denote pixels for which the range value is totally erroneous (outliers often correspond to weak lidar returns for which it is difficult to make the desired range measurements). Dropouts (that are recognized as such by the sensor) and outliers (that must be found via image processing) are sometimes referred to collectively as “missing values (MV).” The authors use a special numerical value, known as “nil,” to represent MVs.

As indicated above, the particular type of noise in a range image tends to be highly dependent upon the particular sensor used. In some types of range images, the presence of outliers and/or dropouts is visually obvious. In the case of the TSLR data, outliers and/or dropouts were not immediately obvious, but casual examination of the range image values revealed that the outliers were quite numerous. However, it is not until one began to exercise the reprojection algorithm on the TSLR images that one got a good understanding of the nature of these particular outliers. Indeed, some vertically-reprojected images (which are intended to produce overhead views of scenes) clearly showed two distinct copies of each reprojected target (and in fact of about every feature in the image). The copy closer to the sensor always appears stronger (i.e., more “populated”) than the other (which can be called the “echo”), and the distance between the two copies appears to be a constant. This finding was also made independently by Alliant Techsystems in a report that was recently made available to the authors [8].

Producing a definitive explanation of the source of this “ghost” artifact in TSLR data is hampered by a lack of available information regarding the TSLR sensor and the method used for generating the fine, absolute range images. The authors’ tentative explanation is as follows (see Figure 6). It appears that the TSLR sensor consists of an FM channel that produces coarse, absolute range values and of an AM channel that produces fine, ambiguous range values. From the Alliant report, the ambiguity interval of the AM channel was found to be 18.75 m (the wavelength of the modulating signal imposed on the ladar beam). With reference to Figure 6, assume that the fine, ambiguous range value produced by the AM channel is about 7 m. This means that the true range value could be 7 m plus any multiple of the ambiguity interval. All possible true fine-range values corresponding to the AM channel measurement are shown as a train of down-pointing arrows separated by 18.75 m. If the coarse-range value produced by the FM channel is as indicated by the down-pointing, solid arrow labelled 1, then it seems logical that the fine, absolute range value corresponding to the combination of the AM and FM measurements would be as indicated by the up-pointing, solid arrow labelled 1. Of course, this presumes that the combined AM-FM range is obtained by selecting the possible true fine (AM) range that is closest to the given coarse (FM) range. (But it is not known whether this is the criterion used in producing the images discussed in this report.) If the coarse-range value produced by the FM channel was erroneous (perhaps as a results of system instabilities) and represented by the down-pointing, dashed arrow labelled 2, then the fine, absolute range value would be the one indicated by the up-pointing, dashed arrow labelled 2. Therefore, even slight changes in the coarse (FM) range measurement can create a jump of 18.75 m in the combined AM-FM range. These possible jumps are consistent with the presence of a true target and a target echo 18.75 m behind it. (The *measured* separation for the example of Figure 6 is 18.58 m.) As indicated earlier, every feature in the image is likely to exhibit this ghosting effect. Close examination of vertically-reprojected images indeed shows that foreground and background features (the main features besides the targets) also have ghosts.

Clearly, cleaning out these particular TSLR outliers is very specific to the TSLR data and may not be relevant to the actual UGV/RSTA sensor. All the experimental results shown in this report were produced *without* attempting to correct the “ghost” artifacts. However, shortly before Demo B (June 1994), the authors decided to explore whether the failure of the HLV algorithm (described later) on a few isolated images was actually due to “ghost” artifacts. Based upon their tentative understanding of this artifact, the authors experimented with a very simple “degghosting” algorithm (quite different from the one proposed in the Alliant Techsystems report). The principle of the algorithm is as follows. A window (typically of size 3×3) is placed around each pixel in the raw range image and the window is then searched for pixels that are 18.75 m in front of the center pixel (plus or minus some tolerance). If the number of such pixels is greater than some fixed threshold value, the center pixel is declared to be a ghost and its range is decreased by the ambiguity interval. By experimenting with the size of the window, the tolerance, and the threshold value, a very simple and effective algorithm for degghosting TSLR images was quickly found. In fact, how much of the ghosts were brought back on their corresponding pattern can be controlled to some extent.

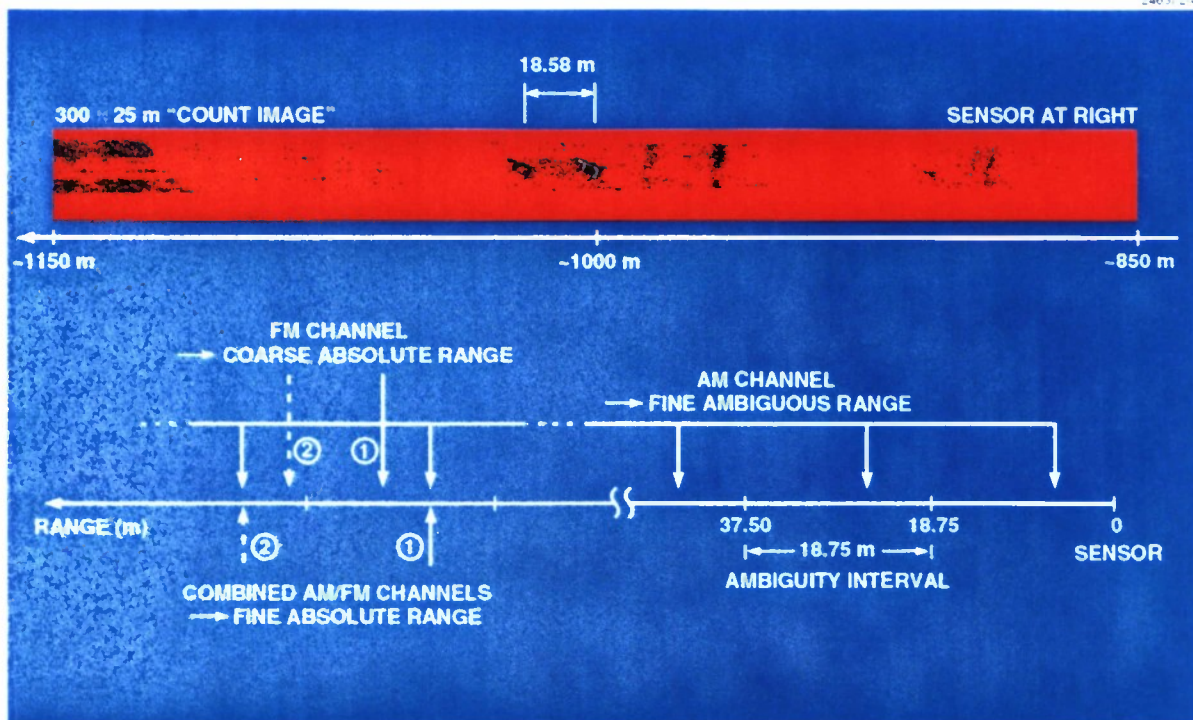


Figure 6. Probable source of "ghost" artifact in TSLR data.

It is interesting that deghosting did not lead to any visible improvement of the HLV algorithm. (The few failures reported above were traced to other abnormal image degradations). However, the capability of eliminating the ghost from overhead views of targets is significant for the recognition step.

In any case, the TSLR data is a good example of data where a-priori information on the nature of the noise and the sensor can be exploited for cleaning the raw range data.

Once again, all experimental results presented in this report have been obtained without the use of the deghosting algorithm. In the future, the deghosting algorithm will most probably be integrated in the preprocessing step (mainly to improve the appearance of overhead views of targets) when processing TSLR data. But it is likely that this algorithm will be irrelevant to the UGV/RSTA ladar.

3.3 Detection

3.3.1 Overview

As explained before, the detection process is tasked with finding areas of interest in the ladar images that are passed from the higher-level detection modules that operate on wide-field-of-view images, most probably FLIR images. In fact, the higher-level detection module, the currently-discussed detection process, and the extraction process to be described in Section 3.4 can be viewed as a hierarchy of “focus-of-attention” mechanisms that lead from a wide-field-of-view image to a “cut-out” of a target or of a target-like object.

The detection process is based on a canonical approach that is highly modular and flexible [10]. The central idea is to construct “interest images” that are intended to highlight the areas of the input image that are likely to contain a target of interest. The generality of this approach comes from the fact that one can generate as many individual interest images as one might find useful, either from a single imaging modality (such as range) or from multiple imaging modalities (such as range and intensity, and possibly even FLIR). The main reason for which these images can be used in concert is that they are all pixel-registered and that they use a common scale of interest, typically in the range [0,255].

In their first use, individual interest images are combined in a unique “combined-interest (CI) image” that highlights “interesting areas” in the original images.

In their second use, interest images are used to segment the image into range segments (RS) that contain pixels that are approximately at the same range and have high concentrations of interest. This use complements the previous one by effectively finding “interesting ranges.”

The CI image and the RS image can then be used jointly to locate areas of high interest. Note that these areas are localized both in the (x,y) plane of the image and in the range dimension. The net outcome of the detection step is a set of RS windows,” where each RS window is a rectangular frame that surrounds a candidate target in a particular RS. The rectangular box that specifies the location of the RS window in the (x,y) -plane will be referred to as the “window.” The sizes of such windows are directly related to the maximum height and width of the targets of interest and to the average range of the RS of interest. Note that the window is intended to extract corresponding subimages from any of the imaging modalities (even if the imaging modality of interest was not used in the detection step).

The idea of computing and combining interest images is a powerful, general-purpose mechanism for performing fusion at the pixel level [10]. It has been used successfully in a number of recognition systems developed and tested at Lincoln Laboratory.

The range-segmentation technique is, of course, tailored to range imagery, but it is also general in the sense that it can use any number of individual interest images, just as is the case for the calculation of the CI image. Note that the range-segmentation technique is, in fact, general purpose and could easily be adapted to a dimension other than range, such as velocity.

In the present approach to detection, all interest images are used intentionally in a rather uniform fashion (to compute both the CI image and the RS image). As a result, the particular set of interest images to be used can be adjusted dynamically and/or new interest images can be added at any time. One possible use of this last feature for the UGV/RSTA project is the later inclusion of the FLIR imagery in the detection process just described. In that case, it is particularly interesting to note that the FLIR imagery would, in fact, contribute to the segmentation both in (x,y) space (through the CI image) and in range (through the RS image).

3.3.2 Principle

Interest Images. Based upon the authors’ prior experience with range imagery, a number of interest images are believed to be useful in the detection process [3, 10, 11].

- **height-limited verticality (HLV):** A verticality image attempts to highlight surfaces that are vertical with respect to some reference plane. Of course, such an image would highlight targets as well as telephone poles and buildings. An HLV image is intended to find vertical surfaces, the height of which is within specified limits (e.g., in a narrow range around 2.0 m for typical tactical military vehicles). Therefore, the higher the value at a pixel in a HLV image, the most likely the pixel belongs to such a surface. A well designed algorithm should work properly for all possible ranges to a vertical surface and the height constraints should be expressed in units of meters (not in numbers of pixels). Further processing details are given below.

- **size-limited blobs:** Targets and other objects sticking out of the ground, whether vertical or not, tend to stand out as darker (closer) blobs in range imagery. Blobs within a range of vertical and horizontal sizes can be extracted from range images using a technique that was previously developed and called “adaptive grey-scale mathematical morphology (MM)” (for range imagery) [12, 13].
- **rods:** The antennas, guns, and other straight, elongated appendages often found on military vehicles can also be exploited using MM techniques. Even though the small diameter of these features corresponds to only a fraction of a pixel at longer ranges, it is significant that these rod-like features often continue to show up with a full one-pixel width at these longer ranges (it appears that this phenomenon occurs in time-of-flight ladars, but not in AM-based ladars). Hence, these thin features should be exploited in the detection step.
- **intensity:** Intensity should not be ignored as a potential interest image, either raw or after some processing. Most CO₂ ladar intensity images are rather poor, but this may not be the case with the UGV/RSTA sensor because it will operate at a more favorable wavelength.
- **FLIR:** Even though exploiting the FLIR imagery is not part of the current primary task, the possibility of constructing interest images based upon FLIR should not be ignored.

Height-limited Verticality (HLV) Interest Image. Because of the desire to achieve an end-to-end system as quickly as possible, it was decided that the detection process would initially be based on a single interest image, which then trivially becomes the “CI image.” As progress is made, further interest images will be created and combined with the previous one(s) in a systematic manner. The first interest image to be used is the HLV interest image, which is believed to be the most promising of all potential interest images (it is certainly the one that best exploits the nature of range imagery).

As indicated earlier, HLV interest images are intended to highlight surfaces that are vertical and have a range of preferred heights. A roughly planar surface is said to be vertical if the plane that most closely approximates that surface is perpendicular to some reference plane. Currently, it is assumed that the sensor and the target are located on local reference planes that are either coincident or parallel to each other. If appropriate additional information were available (such as sensor attitude and ground orientation in the vicinity of the target), the algorithm could be modified to find surfaces that are vertical with respect to target ground, even though these surfaces might appear tilted in the sensor image(s).

Note that the HLV algorithm is designed to deal with MVs, i.e., with pixels that are known to be either dropouts or outliers. The advantage of this capability is that it is only necessary to detect MVs (more specifically, one does not need to find replacement values for them, with the advantage that no assumptions are made about what these replacement values should be). However, this capability has not been exercised so far because the cleaning step is currently bypassed and none of the pixels are marked as missing.

Combination of Interest Images. As previously indicated, each interest image is required to be normalized to a common scale of interest ranging from 0 to 255. A number of methods can be used for combining the various interest images into a single CI image, which is also an interest image (with values in the range [0,255]). In all cases, the combination is done independently at each pixel. The most straightforward method is to simply average all interest images. Another approach is to find the maximum at each pixel across all interest images: this is equivalent to a “fuzzy-or” of all interest images. More complex methods could be implemented if necessary, such as a weighted average of all interest images.

A major advantage of the approach is that newly created interest images can easily be added to the system by simply including them in the standard combination operation (without having to change any other part of the system). Therefore, one can start with a single interest image (e.g., “HLV”) and then, as the need arises, one can incorporate additional interest images such as “size-limited blobs,” “rods,” “intensity,” “FLIR,” etc.

Range Segmentation via Interest Images. Interest images can also be used to segment the range imagery. The proposed approach is as follows. A range histogram is produced for each individual interest image, where the value at each bin of the histogram is the integrated (summed) interest for all the pixels that belong to that particular interest image and are at the range corresponding to the bin of interest. The histogram bins are ranked according to their values (of integrated interest) and the ranked histogram bins are merged by straightforward averaging. The highest ranking bins in the combined histogram are found and the range of each such bin becomes the “center” range of an RS. The corresponding RS consists of all the pixels, the ranges of which are close to this center range. Each high ranking bin that is retained produces an individual RS. The net outcome of the range segmentation process is a RS image where the pixel value is the rank of the RS the pixel belongs to (some of the pixels may not correspond to any of the retained segments).

Range-Segment (RS) Windows. The creation of RS windows relies on the simultaneous use of the CI image and the RS image. The CI image delineates interesting areas in the images and the RS image is used to ensure that interesting areas at clearly separated ranges are not mixed.

The current plan is to begin by thresholding the CI image at a fixed level and to extract all regions above threshold (setting the threshold is relatively simple because the CI image values are always in the range $[0,255]$). Then, the binary map of the regions above threshold is intersected with the binary map corresponding to each of the RSs taken individually. The result is also a binary map of regions that correspond to the parts of the original images that are interesting *and* at approximately the same range. Because this map may contain many fragmented and/or small regions, some cleaning is performed (note that the notion of smallness is expressed in absolute units of meters and converted to units of pixels according to the range of the RS of interest). Fragmented regions that are in the vicinity of each other are regrouped, and remaining, small, isolated regions are eliminated.

All these steps are conveniently implemented using binary mathematical morphology. All remaining regions are extracted and labelled using a classical connected-component algorithm. The process is then repeated for all other RSs. The result is a number of (possibly overlapping) “islands of interest”. (The islands can overlap because the morphological cleaning is applied independently to each RS.)

Then, the interest values in the CI image are integrated over each of the labelled islands and the integrated interest is used to rank these islands in order of decreasing integrated interest. The N highest ranking islands are kept for further processing, where N is a number that depends upon a number of factors, including the computational resources available.

Having determined where the islands of interest are, one can now place “windows” around each of these regions. The horizontal and vertical size of each window is a function of the average range for the corresponding RS and of the maximum width and height of the targets of interest. The x and y sizes of these windows will vary with range and are intended to fit snugly around each candidate target whatever its range (thus, a small window will frame a distant target in an image and a large window will frame a close target in the same image).

The centering of the windows on each island of interest can be performed in several ways. One approach is to center each window on the pixel of the island of interest that has the highest value in the CI image. Another is to use the centroid of the island. In either case, the final position of the window may have to be adjusted so that the whole window fits inside the image frame. All subsequent processing will take place inside each of these windows and the processing in each window will be performed independently. One uses the term “RS window” as a reminder of the fact that, even though the “window” appears to be rectangular, only its intersection with the corresponding RS is the area of interest.

There are a number of ways in which the processing in each window can be controlled. For example, one can decide to process all windows in all cases (most certainly in the order of decreasing integrated interest). Or, one can decide to start with the highest ranking window and only proceed to the next window if no satisfactory recognition was achieved in the first window (and so on, down the list of RS windows). This orchestration will be provided by higher-level control.

As previously explained, the 2D window associated with a particular RS window is a rectangular box that can be used to extract a subimage from any image that is pixel-registered with the original sensor image(s). It is these subimages (e.g., range, intensity, FLIR, HLV, etc.) that are subjected to further processing beginning with the extraction step in Section 3.4.

3.3.3 Status

In order to implement an end-to-end system as rapidly as possible, some of the detection steps have been skipped temporarily.

Because most of the intended experiments were to be based, either on the TSLR data with the available “B” (0.1 mrad) and “C” (0.05 mrad) resolutions, or on their synthetic equivalents, it was felt that the need for using RS windows was reduced (in part because each image was known to contain only one target). Therefore, RS windowing was skipped in all cases. In effect, the sizes of the windows were assumed to be those of the sensor’s images, and the true range to the target was used whenever an estimate of the target range was needed (the target range for all of the images used so far was about 1000 m).

This assumption worked out perfectly in the case of the 0.05-mrad data (where the target occupies a significant portion of the image), and relatively well in the case of the 0.1 mrad data (where the target occupies a smaller portion of the image). The fact that problems (described later) were encountered with the 0.1-mrad data is experimental proof that the creation of RS windows in the detection step is necessary to successfully focus the attention of the system using all the available evidence.

A major element of the detection stage is the HLV algorithm that is believed to provide the most significant individual interest image. This algorithm has been implemented under Khoros and thoroughly tested on several hundred images. Some of the other algorithms discussed above have been implemented and tested for systems built in the past [3, 4, 11] but they will have to be recoded to run under Khoros and to be adapted to the UGV/RSTA application. However, the particular method described above for combining the CI image and the RS image is new and will need to be evaluated.

3.4 Extraction

As explained before, the extraction process is meant to be individually applied to each of the “RS windows” produced by the detection step and scheduled for further processing. While detection is the first mechanism for focusing attention (in an RS window, within an image), extraction is the second mechanism for focusing attention (in a target region, within an RS window). Extraction should result in a “cut-out” of the target from the background. Extraction operates only on the part of the RS window that corresponds to the RS associated with that RS window, and essentially “peels off” the target from the windowed RS. Note that the cut-out shape can be used as a cookie-cutter to extract, without any loss of information, cut-outs from any of the imaging modalities.

The extraction algorithm used in the system described in [11] is a natural candidate for the present application. However, it will have to be reimplemented under the Khoros environment.

However, there is a possibility that the extraction step might be bypassed as a result of the use of FTs in the recognition step. The FTs might indeed be well suited to perform both extraction and recognition in one step. In addition, the focusing of attention performed in the detection step above (with the creation of the RS windows) might be sufficient to reject much of the clutter.

In spite of the above comment, all or part of the extraction step might still be needed in order to estimate the position of the ground in the vicinity of the candidate target. (Ground estimation is an integral part of the extraction algorithm discussed in [11].)

At this point in time, none of the planned extraction algorithms have been implemented, but they will be implemented if the need arises.

3.5 Recognition

3.5.1 Overview

As explained before, the recognition process is intended to provide the most likely identity of the candidate target possibly present either in a “RS window” (if extraction is skipped) or in an extracted “cut-out” (if extraction is performed).

The general approach is as follows. First, an overhead view is produced corresponding to the scene of interest, i.e., either within the RS window or the cut-out (the different types of views that can be produced are discussed below). Second, a rough estimate of the target orientation is found by using a HT designed to look for straight lines (the sides of the targets in the overhead views). Third, FTC with azimuth-dependent FTs is performed to determine the most likely identity of the candidate target, its pose (position and refined orientation), and a degree of match that quantifies the degree of confidence the system has in the identification performed. Note that the role of the HT is mainly to reduce the angular search performed with the FTs.

3.5.2 Principles

Vertical Reprojection. The key piece of information that makes the proposed approach to recognition feasible is the range image provided by the ladar sensor. Indeed, the knowledge of (a) the absolute range value at any given pixel, (b) the x and y indices of that pixel, and (c) the x and y angular sizes of each pixel, allows one to determine the 3D position of the point in the scene that was mapped into that particular pixel and with that particular range value. Once the 3D world position of each pixel in the image is known, arbitrary geometric transformations can be performed on the range image or any other image that is pixel-registered with the range image. One of the simplest transformation is the vertical reprojection that projects each 3D point vertically to some common reference plane (which is generally distinct from the “horizontal plane” of the world coordinate system). The net result is an overhead view of the scene. However, by using more general transformations, it is possible to produce images of scenes viewed from arbitrary viewpoints. For example, if the orientation of the candidate target were known (the HT and the FTC described below will provide this information), the original imagery could be reprojected into front views and side views, perhaps allowing some straightforward matching independent of target orientation. It should be emphasized that the reprojection operation does not create information that is not part of the original images. Hence the parts of the objects that are not visible in the forwardlooking view will not appear in the overhead view, and this overhead view will appear to be incomplete. (This is further discussed below.)

Very useful overhead views can be made by reprojecting just the range image. However, as suggested above, any image that is pixel-registered with the range image can also be reprojected into an overhead view. Therefore, in the case of the UGV/RSTA sensor, the intensity image could also be reprojected into an overhead intensity view, and, if the appropriate portion of the FLIR image could be brought into registration with the ladar imagery, an overhead FLIR view could also be produced.

Although the concept of reprojection is straightforward, the actual implementation details are not. The following points highlight the delicate nature of the operation.

1. The appropriate 3D world frame must be selected.
2. The local frame for reprojecting the scene of interest (e.g., an RS window or a cut-out) must be selected in a judicious fashion if one wishes to keep the size of the reprojected image small and be sure that the target of interest will be reprojected within that image. Of course, one can always make the reprojected image as large as necessary to bring the reprojected target in view within the reprojected image. The problem (discussed further below) is due to the fact that even a small forwardlooking lidar image covers a lot of terrain, from a few tens of meters in the foreground to a few thousands of meters in the background. The selection of the local reference frame (the origin of which will correspond to the center of the reprojected image) would be relatively simple if the RS window was always centered in the original image. However, this is almost never the case, and the local reference frame position and orientation must be automatically adjusted to the actual position of the RS window within the image.
3. The pixel sizes of the reprojected image must be carefully selected. These sizes (in units of meters) need not be matched exactly to the angular pixel sizes (in mrad) of the original imagery. A variety of pixel sizes were tried, ranging from 0.05 to 0.25 m, but no particular value has been selected so far (0.10 m is often used with the 0.05 - mrad resolution images). The tendency to lose the target in a small reprojected image (say 200×200 pixels) with a 0.10-m pixel size is easy to understand. Indeed, this is equivalent to looking at possibly several square kilometers of terrain through a 20×20 -m straw. Without appropriate control of the geometry, the probability of seeing the target in the reprojected image is very small. If the sizes of the reprojected image are increased (most probably in the downrange direction to cover more terrain), then the burden placed on the FTC step that follows becomes prohibitive. Hence, the need for proper RS windowing and choice of reprojection reference frame.
4. One can increase the likelihood of placing the reprojected target within the reprojected image by adjusting the position of the reference frame of the reprojected image according to the location of the cut-out (instead of the RS window). Whether the RS window or the cut-out is used, the following, additional mechanism, which was implemented, is useful for placing the reprojected target as close as possible to the center of the reprojected images. This automatic mechanism actually centers the reprojected image either on the average, mode, or median of the x and y coordinates of all the reprojected pixels. (As one might expect, the mode is by far the best mechanism.)

So far, no effort was made to precisely define which set of pixels should actually be reprojected. In fact a number of choices are possible. They are listed below in order of tightness from the least tight to the most tight.

Whole image: Of course, images can always be reprojected without computing RS windows or cut-outs. In fact, because much of the windowing mechanism is not currently implemented, this is what is currently being done with the TSLR data and with the corresponding synthetic imagery.

Window: Here, one reprojects all the pixels inside the rectangular area defined by the RS window. This has the advantage of eliminating all the distracting clutter outside the window. Note that the RS that supported the construction of the RS window is not totally ignored, since its average range is used to correctly position the local reference frame of the reprojected image.

Range-segment (RS) window: Now, only the pixels that are both in the window and in its associated RS are reprojected.

Cut out: In the tightest of the cases, only the pixels in the cut-out are reprojected.

The main danger with the last two cases is the loss of some of the pixels along the edges of the targets. Therefore, it may be more appropriate to reprojects all the pixels in the (rectangular) window, and to let the FT-based recognition stage sort out the target from the clutter. To simplify subsequent discussions, it will be assumed that the pixels in the window are the ones selected for reprojection. However, the discussions carry over to the other possible sets.

Four important overhead views of scenes and objects can be produced by using just range imagery. Before starting the reprojection process, it is convenient to initially mark all the pixels of the reprojected image with a special numerical value that is referred to as “nil” or “missing.” (This is the same special value as the one previously discussed for marking missing values in range images.) Even though this may not be necessary in the first two cases below, this special value becomes necessary in the last two cases.

Trace image: Each pixel in the window is reprojected and the reprojection pixel it falls into is marked with a “1”. Once all pixels have been processed, the remaining nil’s, which can be quite numerous, may optionally be turned into “0”’s to create a binary image. In any case, this “trace image” shows where the visible features of the object fall in the reprojected image. In spite of its simplicity, this image can be quite useful to deal with “obscuration from below,” as would be the case for a tank partially hidden behind a stone wall. The reason for this will become clearer as the next case is discussed.

Count image: Each pixel in the window is reprojected and the value of the reprojection pixel it falls into is modified as follows. If the pixel value was nil it is set to “1”. Otherwise the pixel value (the “count”) is incremented by one. Here also, the remaining nil’s may optionally be set to zero. In any case, the resulting “count image” shows, not only where the features of the object fall into the reprojected image, but also the “height” of vertical surfaces. To illustrate, consider an object consisting of a vertical column that has a height of N pixels in the original sensor image(s). The reprojected count image will consist of a single nonnil pixel with a count of N . Of course, that particular pixel will end up near the base of the column in the reprojected image. Observe that the count is not always a measure of absolute height. It is only a measure of the vertical extent of vertical surfaces (which may or may not touch the ground). The “count image” acts somewhat like an edge detector. Note that the count image may be difficult to interpret in the case of “obscuration from below.” Going back to the example of a tank behind a stone wall, the counts will not be uniformly decreased by the height of the wall. This will only be the case for columns of pixels that are actually touching the ground (on which the wall is assumed to be standing). This limitation is one of the reasons for which the “trace image” discussed above may be useful. In addition, special care must be taken to correctly normalize the count images so that the effects of varying angular pixel sizes and range can be taken into account.

Z-coordinate image: Each pixel in the window is reprojected as in the previous cases, but the value of the reprojection pixel it falls into is set to that of the Z coordinate of the 3D world point corresponding to the pixel being reprojected. Note that the Z coordinate of a specific point on a target will change drastically as the sensor *and* the target move up and down on the terrain, and as the world coordinate frame *and* the reprojection frame are dynamically adjusted. In fact, the Z -coordinate image is nothing more than a height-above-ground image with an unknown bias that is the same for all the pixels. Clearly, the remaining nil values cannot be converted to some other arbitrary value (such as “0”) because this arbitrary value might actually be a valid Z coordinate!

Height-above-ground image: The Z -coordinate image could be transformed into a true height-above-ground image if the Z coordinate of the terrain under or near the target could be determined. Whether or not this can be done in a reliable fashion remains to be determined. However, the extraction algorithm that is being considered includes a built-in mechanism for estimating the shape of the local ground near the target [11]. Alternatively, ground estimation systems developed by others may provide the information that is needed to create the height-above-ground image. Note that the height-above-ground image (as well as the Z -coordinate image) do not suffer from the scaling problem associated with the count image.

The VR algorithm is used in two distinct fashions.

1. VR is applied off-line to ideal, noiseless, unquantized, purely geometric synthetic range images of isolated target vehicles in order to guide the design of FTs. (Note that images other than range could also be used for this purpose, e.g., FLIR.) The main reason for considering isolated targets without any kind of backgrounds under or behind the vehicle (thus considering vehicles that are effectively floating in space) is based on the observation of a number of unrealistic artifacts that appear when low-depression-angle images are reprojected into a vertical view. Indeed, pixels (in the original images) that correspond to the ground surface near the target fall (in the reprojected images) on parallel lines perpendicular to the line of sight and separated by several tens of meters. In most of the ideal reprojected images one tried to create, the ground appeared as a single line running horizontally through the center of the image (with the other ground lines falling beyond the top and bottom boundaries of the reprojected image). Planar walls of any kind would also lead to unrealistic configurations of parallel lines. These observations were the basis for deciding that the ideal synthetic images used for designing FTs should not have backgrounds of any kind.
2. VR is also applied to real sensor imagery at mission time (the part of the sensor imagery that should be reprojected was discussed earlier, and is typically one of the detection “windows”). As indicated later, the vertically-reprojected images are used in two different ways at mission time: (1) for target orientation estimation via HT, and (2) for identification and pose estimation via FTs. Of course, it is important that the pixel sizes of the reprojected image(s) be identical to those of the FTs used for recognition. However, the pixel sizes of the sensor images and those of the synthetic images that are created to design FTs (either automatically or manually) are not required to be identical (although it is probably a good idea to make them similar). At the present time, all synthetic images that have been produced have pixel sizes of 0.05×0.05 mrad, whereas the highest-resolution TSLR images have pixels sizes of 0.0397×0.0358 mrad (see Table 1). And the common pixel sizes that are used most often for the TSLR and synthetic reprojected images are 0.10×0.10 m.

Note: It is very important that the correct pixel sizes (also known as pixel distances or as sampling intervals) be used for the image that is being reprojected. Failure to do so will lead to distortions in the reprojected image. (However, one danger is to believe that the data is actually responsible for the distortion and never to realize that the distortion is actually due to an incorrect choice of pixel sizes!) Significant distortions will appear in the 0.05-mrad resolution TSLR images when the pixel sizes are erroneously believed to be 0.05×0.05 mrad, when they are actually measured to be 0.0397×0.0358 mrad. The most noticeable distortions are lines that appear at the wrong angles and angles that become wider (with target outlines being no longer rectangular). The incorrect choice of pixel sizes easily explains all these deformations.

Now, one can discuss the two algorithms that use vertically-reprojected images as part of the recognition step.

Hough Transform (HT). The goal of the HT is to find the orientation of the long axis of each candidate target. The main reason for using the HT is to reduce the computational burden associated with the FTC operation that follows. If the HT were not used, the FTs would have to be spun 360° at each pixel in the reprojected image. Because the HT applied to the reprojected image appears to be capable of providing a good estimate of the target orientation (except for front and rear views, as discussed below), the angular range over which FTs have to be spun can be reduced considerably, maybe reducing the FTC computational burden by a factor of 5 to 10.

Experiments show that the “count image” (one of the possible reprojection images discussed earlier) is particularly well suited as an input to the HT. The reason is that the count image acts as a kind of edge detector that highlights the edges of the vehicles. In most cases, the sides of the vehicles (as opposed to their fronts or rears) tend to correspond to the main peak of the HT. However, in the cases of near-head-on and near-tail-on situations, the front edge or the rear edge of the vehicles almost always correspond to the main peak of the HT (of course, this is an undesirable effect because the orientation estimate will be off by 90° or so). All other reprojected images discussed earlier (e.g., the height-above-ground image) exhibit fairly uniform blobs that are not well suited for finding orientation through the HT. In addition, note that the use of the HT to find the target orientation in the count image is quite insensitive to potential errors in the scaling of the count image values (of course, such errors should not arise in a well designed system) and to “obscurations from below” (such as by the stone wall previously alluded to). Each of these problems could create potential difficulties when the FTs are applied to the count image. Hence, the HT offers a chance to get a preliminary estimate of the target orientation without having to worry about these problems.

The HT can find the general orientation of the target, but cannot tell where the front of the vehicle is pointing to. Therefore, there is always a 180° uncertainty in the orientation estimate provided by the HT. This means that the FTs must be spun in two distinct angular sectors separated by 180° . The use of FTs should allow us to determine the true azimuth angle of the vehicle (i.e., without any 180° ambiguity).

In practice, it was found useful to apply the HT to a thresholded version of the count image, thereby ignoring isolated pixels that often correspond to outliers. The value of the threshold that is used routinely is around 1 or 2.

Peak Detection. The orientation of the target is found by locating the highest local peak in the 2D array that corresponds to the HT. This is done by probing each pixel in the (properly expanded) HT with a square window of selectable size (typically 3×3) and asking whether the pixel being probed is larger than any other value in the window. If it is, the pixel in question is declared a local peak. The “highest” of the local peaks is then used to derive the orientation and radial position of the visible side of the vehicle. As is typical, the angle obtained is in the range from -90° to $+90^\circ$ (excluded), thereby exhibiting the anticipated 180° ambiguity. So far, no attempt was made to use the radial position of the side of the vehicle, but this is potentially useful information (e.g., for reducing the spatial search carried out with FTs).

Squished Hough Transform (HT) The classical HT can be described as a 2D array with the horizontal axis corresponding to angle and the vertical axis corresponding to (radial) position. One has also investigated the possibility of “squishing” the standard 2D HT into a new 1D array (which is referred to as the 1D HT) by collapsing all the values in each column (corresponding to a particular angle) into a single number. A number of operations were investigated including “minimum,” “maximum,” and “average.” The averaging operation is particularly interesting. First, it does not take long to realize that straight averaging (or summation) of all the values in each column results in a 1D HT that has a necessarily constant value at all angles. (This is actually a good way to test that the HT and the squishing operation are correctly implemented!) However, simple modifications of the straight averaging method yield interesting and potentially useful results. The two modifications that were investigated are as follows:

1. A threshold is used to effectively ignore any pixel of the 2D HT that has a count below the threshold (for the case of count images). This eliminates many of the pixels that are not on one of the prominent lines in the reprojected count images. (This threshold is distinct from the threshold that is applied to the reprojected images.)
2. The sum (in each column) of the 2D HT pixel values that are above threshold is normalized by the number of pixels that have a nonzero count in that column. This has the effect of boosting the normalized sum at the angles where the main ridge of the HT sinogram is the narrowest.

The related experiments of Section 4 (on synthetic data) indicate that the net result of these modifications is to produce a 1D profile that also has a peak at the target orientation. (Peaks are found using the previously described algorithm, which is designed to work both in 1D and 2D.) The experiments also indicate that this approach narrows the angular range where the HT makes errors of about 90° , i.e., near head-on or near tail-on. The usefulness of this method for real data appears doubtful.

Functional Template Correlation (FTC). The heart of the proposed system lies in the application of azimuth-dependent functional templates to vertically-reprojected scenes that are likely to contain vertically-reprojected views of targets. As discussed previously, the scene that will most likely be reprojected is the full content of each “window” obtained during the detection step.

Before going further, one presents a brief review of FTs and FTC [4, 14, 15, 16].

In conventional image-processing correlation, the correlation kernel is a template containing pixel values. The template is overlaid on an image at successive pixel positions. For a given placement of the template, the template values are multiplied by the underlying image-pixel values and these products are summed or averaged. As the template is swept across the image, the output correlation image is generated one pixel at a time. Correlation peaks indicate the presence of patterns corresponding to the selected kernel. Of course, the correlation process can be repeated at various orientations to look for rotated patterns.

In FTC, the operation combining the pixels that are in correspondence in the image and the template is not limited to multiplication followed by summation (or averaging). In FTC, each pixel value on the template is an index, say n , to a function, say $f_n(x)$. If the value of a template pixel is n , and the value of the corresponding image pixel is I , then the product nI that would be computed in standard correlation is replaced by the function value $f_n(I)$. The accumulation of these values across the template is typically done by averaging followed by clipping. In most applications of FTs, the input image has a fixed range of [0,255] and the resulting correlation is forced to have a fixed range of [0,255]. In these cases, function evaluation can be done by table lookup, which leads to very fast implementations. Just as in the case of standard correlation, FTs are translated from pixel to pixel in an image, and they can also be rotated to look for rotated patterns.

The FT approach is very generic and highly modular. The main algorithm that needs to be implemented is a FTC engine that can be applied to any overhead-view image, provided a related set of FTs is available.

The overhead-view images that could potentially be used as input to the FTC engine are many. One has already discussed a number of overhead views that are derivable from just the range image and are thus highly geometric in nature: the most promising of these images are the “trace image,” the “count image,” and the “height-above-ground image.” The advantages and disadvantages of these images have been discussed previously. In addition to these three images that are based on range only, any other image that is pixel-registered with the range imagery can also be transformed into an overhead-view image. In the case of the UGV/RTSA sensor, the main candidate is an “overhead intensity image” and, possibly, an “overhead FLIR image” (provided pixel registration can be achieved between the FLIR and the ladar). For other sensors, such as the TSLR sensor, an “overhead Doppler image” could also be created.

To perform recognition via FTC, one must design a set of FTs that is specific to the particular type of overhead view one wishes to work with. For example, a specific set of “count FTs” should be created to work with the overhead “count image.” Before discussing the nature of such a set of templates, it should be emphasized that the FTC engine is being designed from the start to deal simultaneously with multiple sets of templates. One way to picture this is to imagine (a) the FTC correlation process going on independently, but in synchrony, in each of the selected overhead views, and (b) the addition of all the scores being performed across the views for each spatial and angular position of each template or template set [4]. Once again, different sets of templates should be used with each particular type of overhead view.

Previous discussions have alluded to a *set* of templates for each particular type of overhead-view image. The reason for this is that the reprojected image of a target will change as the angle (azimuth) of the target changes with respect to the line of sight of the sensor. This change results from the varying self-occlusion of the various target parts as one moves around the target (or as the target rotates with respect to the sensor’s line of sight). Therefore, a reprojected front view of a target will have a very specific appearance, namely only the front features will be visible. Ditto for rear view and side views. Thus, in order to apply FTC to overhead-view images, one must design a set of FTs, where each FT is valid only in a limited range of azimuth angles. In the past, a similar problem was encountered when FTC was used to recognize targets in SAR imagery [5]. In that particular case, good recognition performance was achieved with a set of only 12 templates (five of which were mirror images of five others, because of assumed symmetry) per target. It is believed that a very similar approach will work well in the present application. The way to handle these azimuth-dependent FTs is relatively simple. Indeed, as the templates are spun at each pixel (to look for rotated targets), one simply selects the FT that covers the particular angle being tested. The FTC engine that is being implemented contains the appropriate mechanism for selecting the correct FT in the set.

A particularly convenient approach to begin designing the azimuth-dependent FTs (either manually or automatically) is to use the previously-described ideal synthetic imagery of targets (“floating in space” to avoid ground and background artifacts). Even if these ideal range images can be reprojected in the fashion previously described, it is useful to reproject the synthetic imagery in a reference frame that actually turns with the target. In effect, this takes the rotation out of the reprojected views but leaves all the self-occlusion effects intact. The same approach was used by the authors for SAR imagery [5]. The rotating frame is referred to as the “target frame,” and the conventional, nonrotating frame as the “radar frame.” The reprojection algorithm that has been implemented can reproject any image (range or otherwise) either in radar frame or in target frame (in a single operation that avoids the errors resulting from multiple interpolations). Of course, reprojection into target frame requires the knowledge of the target azimuth angle.

3.5.3 Status

Most of the work performed to date is in connection with the recognition module of this section.

All the necessary vertical-reprojection algorithms discussed above have been implemented (under Khoros) and extensively tested. A reprojection algorithm has also been implemented (under Khoros) for dealing with reprojections from arbitrary viewpoints.

All the algorithms related to the HT (i.e., the HT and the peak detector, both in 1D and in 2D) have also been implemented (under Khoros) and extensively tested.

A number of FTC engines have been implemented in the past in connection with other recognition systems (e.g., see [4]). A new FTC engine is currently being implemented for running in the Khoros environment. This new engine will, for the first time, be able to handle simultaneously (a) azimuth-dependent FTs and (b) multiple overhead-view images. The UGV/RSTA project will directly benefit from various speed improvements the FTC engines have gained over the last few years in connection with other projects concerned with downlooking ladar imagery [4], fully polarimetric SAR imagery [5], and weather-radar imagery [15].

The sets of FTs for the “count image” and other potential overhead-view images have not yet been constructed, but most of the activities related to the creation and reprojection of ideal synthetic images of targets “floating in space” is directed at constructing such sets of templates, initially by hand, and later by automatic means.

The main reason for which the authors have targeted most of their initial work to the recognition module (thereby skipping many of the preceding processing steps) was to convince themselves that it would be feasible to apply FT techniques to overhead views of targets. All the experiments carried out to date in this area indicate that the approach appears to be sound.

3.6 Verification

The goal of the back-end verification step is to apply more complex matching procedures to confirm that the identification made in the preceding step is plausible.

The verification module will not attempt to modify the identification made by the recognition module. “Verification” will probably use the initial pose estimate produced by “recognition” and will probably generate an updated degree of confidence in the validity of the match.

Even though there appears to be ways of dealing with target articulation (e.g., of the turret or gun) at the recognition level (through FTs), it seems that much of the burden of dealing with articulation will be left to the verification level. The reason for this is that the detailed matching one intends to perform in “verification” would not make much sense if the various states of articulations were not explored.

The precise method by which the matching will be done has not yet been determined. One possible approach is to do a direct match of predicted range imagery and observed range imagery (other modalities could also be considered). Another possible approach is to do the matching at the feature level (e.g., using 3D corners). How the matching would be performed in a feature-based approach has not been determined. The candidate methods are:

1. The appearance-model based matching method developed for previous model-based ATR systems (in the various versions of XTRS) [17].
2. The matching method used by the ARAGTAP system developed by Martin Marietta for SAR imagery (which includes the use of Kalman filtering).
3. The method proposed by Fan in his recent book on recognition from range imagery [18].

4. EXPERIMENTAL RESULTS

4.1 Introduction

This section presents some of the experimental results obtained with the algorithms implemented to date under Khoros.

The three main algorithms discussed are

1. The **height-limited verticality (HLV)** algorithm that creates one of the most useful range-based interest images to be used for detection purposes.
2. The **vertical reprojection (VR)** algorithm that (a) creates the inputs to the HT and the FTC engine and (b) provides the reprojected synthetic images necessary to begin the design of the azimuth-dependent FTs.
3. The **Hough transform (HT)** algorithm and its associated **peak detection** algorithm that are used (in either of their 1D or 2D versions) to find the orientation of the target and cue the recognition process based on FTs.

Each of these three main algorithms was applied, at various stages of development, to many of the 764 images (548 TSLR and 216 SYNTH) that are currently available. Virtually all 764 images have been used at one point or another to test one or more of the algorithms. In spite of the many gigabytes of disk space, additional space would be needed to keep all the images that result from applying all algorithms to all available data and with a variety of parameter choices.

Most of the experiments are performed on entire sequences that contain images (primarily fine, absolute range at the present time) of a specific target, at a specific range, and with specific angular resolutions (and pixel sizes). The successive views in a sequence correspond to increasing target azimuth angles starting at 0° (front view) and with angular increments that are close to 5° or a multiple thereof. In the ideal cases where most or all of the azimuths are represented, the sequence is very similar to what would be obtained if the target was imaged while rotating on a turntable. (Of course, the turntable analogy becomes perfect for sequences of synthetic data.)

In the Khoros environment, each sequence can be played as a movie using the “animate” tool. This tool is extremely useful for quickly reviewing the data. Moreover, because each of the implemented algorithms, or sequences thereof, are typically applied to a whole image sequence, the results can also be played back as movies using the same “animate” tool. This is the procedure one routinely follows to look at raw images or result images.

All sequences of interest for the TSLR data are summarized in Figure 1. The naming conventions for the TSLR and SYNTH data are explained in Section 2. The TSLR sequences (data sets) of primary interest in this section are named M35-1000-B6, M35-1000-C6, with similarly-named sequences for the other targets. The corresponding sequences for the ideal, noiseless, unquantized SYNTH images of targets floating in space are M35-1000-010-T, M35-1000-005-T, etc.

Below, whenever applicable, the main algorithms are illustrated by showing the results corresponding to the range images shown in Figure 3 (for real data) and Figure 4 (for synthetic data).

Most of the comments regarding the performance of the algorithms will be of a qualitative nature. However, in the case of the application of the HT and the associated peak extractor for estimation of target orientation, one will present a true quantitative performance evaluation on real data (based on the ground truthing the authors have performed) and on synthetic data.

4.2 Height-Limited Verticality (HLV)

The results of applying the HLV algorithm to the images shown in Figure 3 are shown in Figure 7.

The algorithm was applied to all 548 images in the TSLR data base. Each of the 12 sequences was examined casually using “animate.” In virtually all cases, the HLV interest image fulfilled its role of highlighting the target region. At some point, the “ghost” artifact described in an earlier section was believed to be responsible for the failure of the HLV algorithm on a very small number of images. However, the algorithm continued to fail on the very same images when these were preprocessed with the previously described deghosting algorithm. In the end, the failure was traced to other types of degradations in these particular images.

Each image suffering from the “ghost” artifact can be regarded as consisting of two partially-filled images that are intermeshed on the pixel grid, with one of the images “being” approximately 18.75 m ahead of the other in each local neighborhood. A good analogy is to think of these partially-filled images as incomplete beds of nails that are intermeshed. Retrospectively, it is surprising that the HLV algorithm managed, at all, to highlight the target in the presence of this artifact. What may actually happen is that each of the individual copies of the target is independently found to be vertical. However, the algorithm was certainly not intended to work under these conditions.

In conclusion, the HLV algorithm appears to be quite capable of playing its role of interest image generator in the detection stage.

4.3 Vertical Reprojection (VR)

The results of applying the VR algorithm to nine images of the TSLR sequence M35-1000-C6 are shown in Figures 8 to 10 (in each case, the sensor is located towards the bottom of the page). The “count images” that are shown corresponds to increasing azimuth angles of the M35 truck in increments of approximately 40 degrees, starting with a front view at 0°.

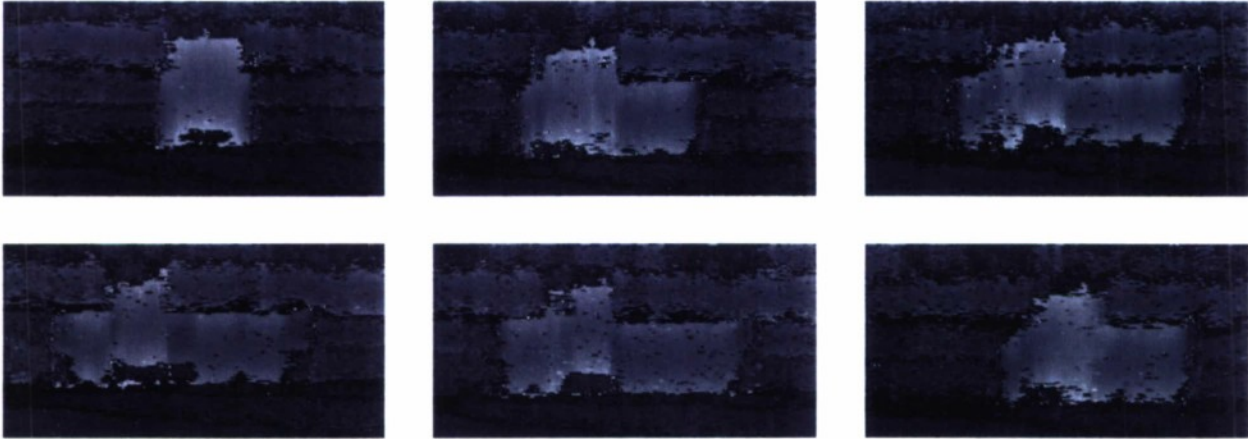


Figure 7. HLV interest images for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad.

Each of the long, skinny images has sizes of 100×1200 pixels, thus covering a strip of terrain that is 25 m wide and 300 m deep (at the 0.25 m pixel size that was used to reproject these images). In each strip, observe the traces corresponding to the foreground (towards the bottom of the strip) and to the background (towards the top of the strip). In between, one can see the target, possibly with its echo, as well as the traces of a number of intermediate hills. Note that the presence of an echo affects not only the target area, but the entire image (foreground and background can often be interpreted as a combination of a main pattern and an echo pattern). Also, observe that each strip is almost perfectly centered on the strongest (and closest) of the target patterns. This is the result of the automatic centering mechanism that is available in the VR algorithm (an evaluation of the centering mechanism is given below).

Also note that the reprojected images shown do not even cover the whole field of view. By increasing the image sizes further and/or by increasing the reprojection pixel sizes, one should ultimately be able to “see” (in the reprojected image) the whole scene present in the original image.



Figure 8. Global overhead view "count images" for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 1). The successive ground-truth target azimuths are 0° , 44° , and 83° .

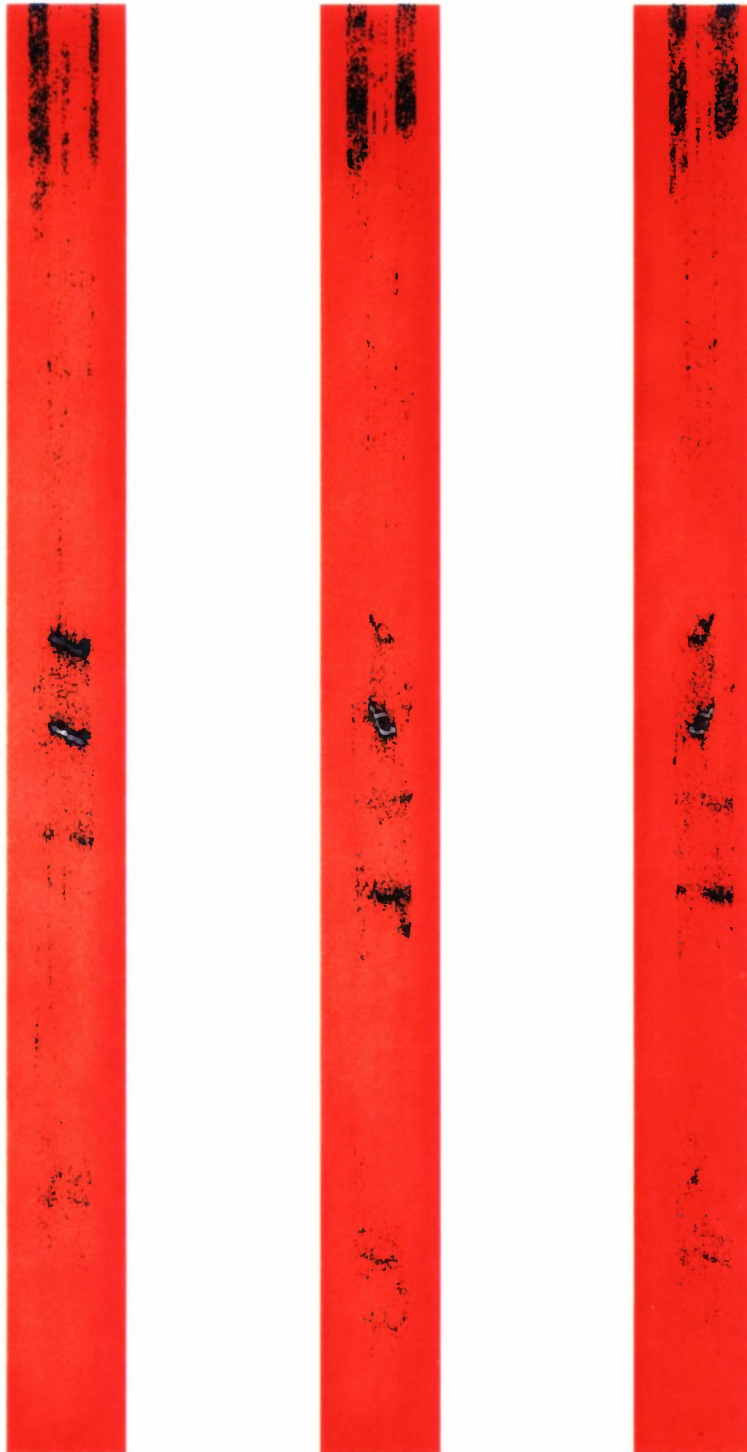


Figure 9. Global overhead view "count images" for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 2). The successive ground-truth target azimuths are 123° , 161° , and 198° .



Figure 10. Global overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.25 m (Part 3). The successive ground-truth target azimuths are 237° , 277° , and 317° .

Figures 8 to 10 are useful to demonstrate the global reprojection capabilities of the VR algorithm and the rich information content of range imagery reprojected over large areas. However, the focusing mechanisms of XTRS-UGV will lead us to consider the immediate surroundings of detected candidate targets. The results of producing “count images” that are more local and have a smaller reprojected pixel size are shown in Figure 11. These images are the results of applying the VR algorithm to the images of Figure 3. They do not necessarily correspond to the long images of Figures 8 to 10. In the present case, each image has a size of 200×200 pixels and a reprojected pixel size of 0.10 m, thereby covering a patch of terrain of 20×20 m.

Here also, the automatic centering mechanism of the VR algorithm was relied upon to bring the target in the center of the frame. The assessment of the automatic centering algorithm applied to whole TSLR images is as follows. Automatic centering worked perfectly in all 285 TSLR images at the 0.05-mrad resolution. But, it was observed to fail in some of the 0.10-mrad resolution images. However, this was totally expected because the targets occupy a smaller portion of the images at this increased angular pixel size. Remember that the plan is to reproject only the pixels inside the detection window, which should fit snugly around the target. Therefore, one expects the automatic centering mechanism to be effective when the windowing stage of “detection” is implemented. Before continuing with the discussion, it should be pointed out that the first images of Figures 8 and 11 (i.e., those corresponding to the front view) were both obtained entirely automatically from the first image of Figure 3. The only differences between the two applications of the VR algorithm were the requested sizes of the output images and of the reprojected pixels.

The VR algorithm was also applied to the SYNTH images shown in Figure 4 and the corresponding (local) “count images” are shown in Figure 12. Figure 13 shows the count images corresponding to nearly identical azimuth angles (near 45°) for real (TSLR) and synthetic (SYNTH) images of an APC, a truck, and a tank.

Figures 11, 12, and 13 indicate that there is significant similarity between the count image of each major TSLR target (APC, truck, and tank) and that of its SYNTH counterpart. Furthermore, Figure 13 shows that there is a significant difference between the count images of the APC, truck, and tank (either TSLR or SYNTH). These observations support the idea of using models (such as FTs) derived from reprojected synthetic imagery to identify targets in reprojected real imagery. The current plan is to use synthetically-produced images, such as those of Figure 12, to guide the design (either manual or automated) of sets of azimuth-dependent FTs.

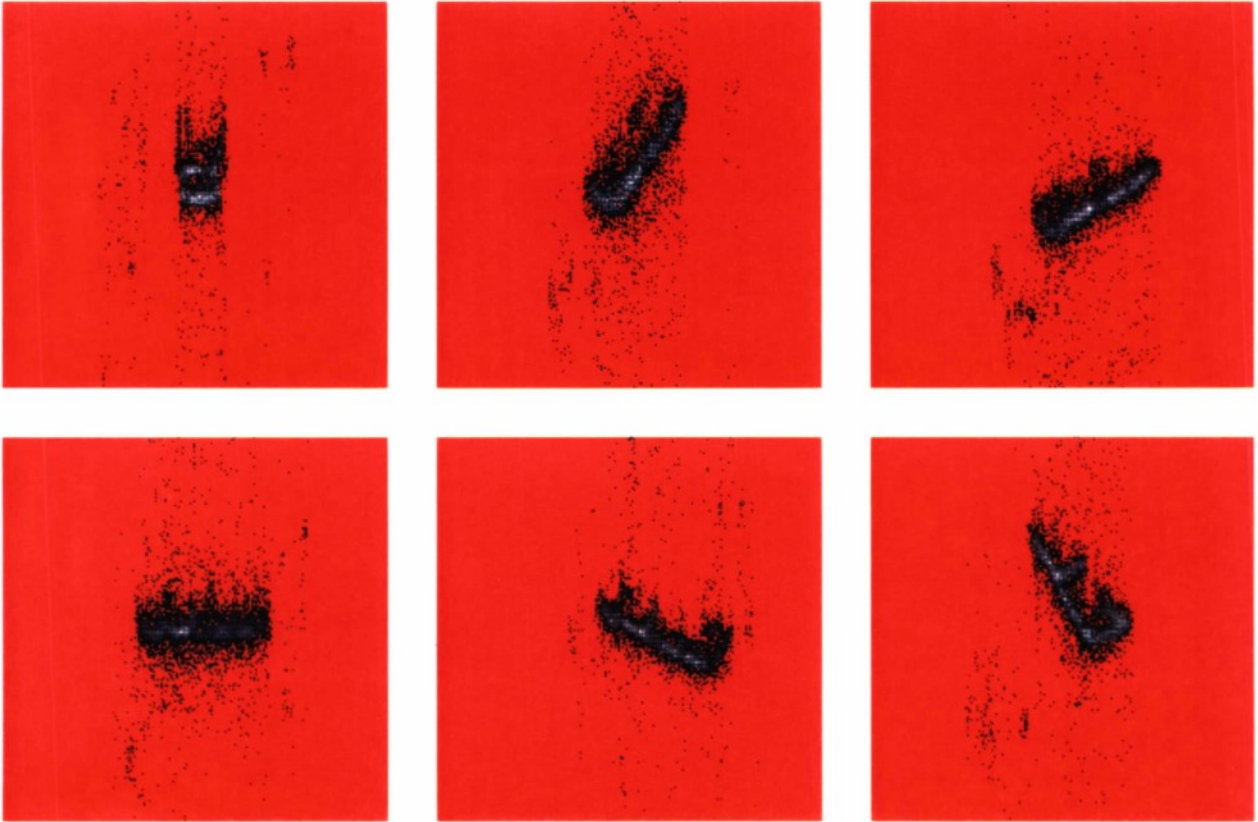


Figure 11. Local overhead view “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.

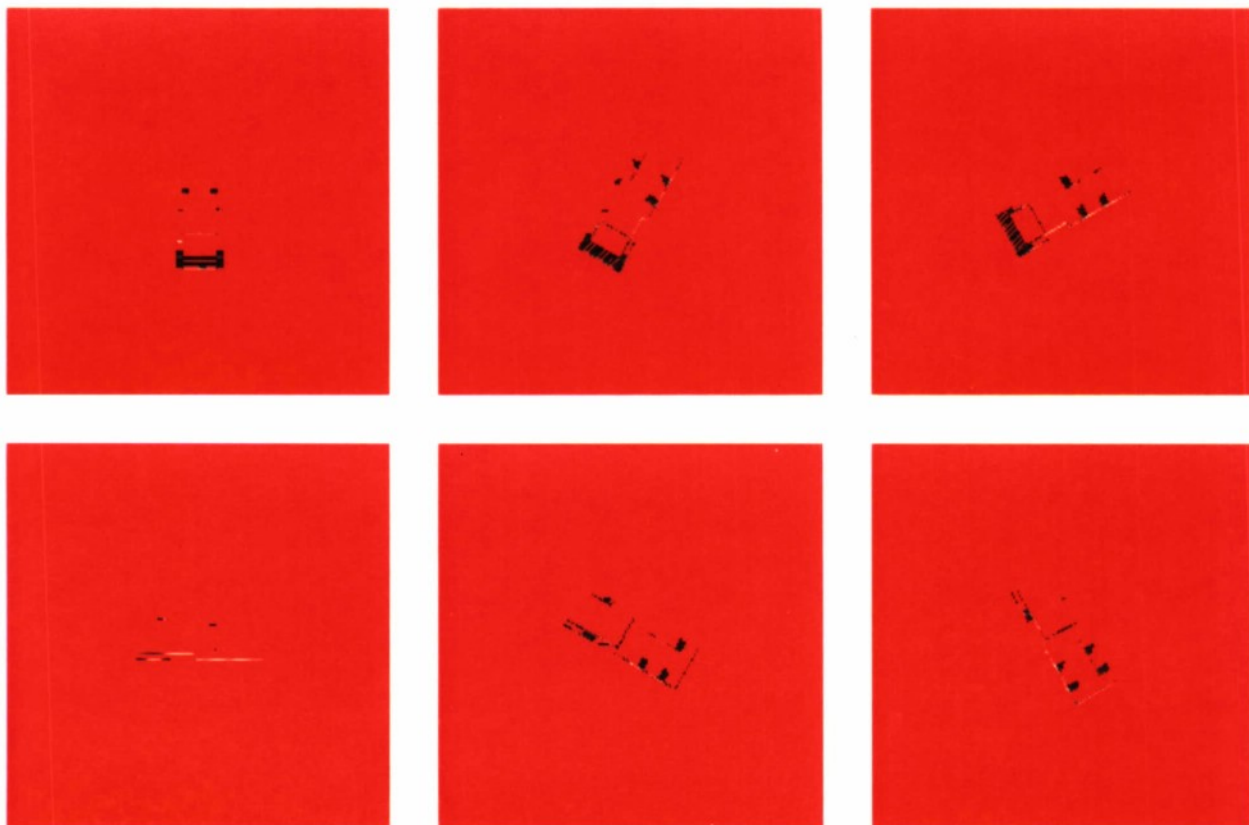


Figure 12. Local overhead view "count images" for the SYNTH M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.

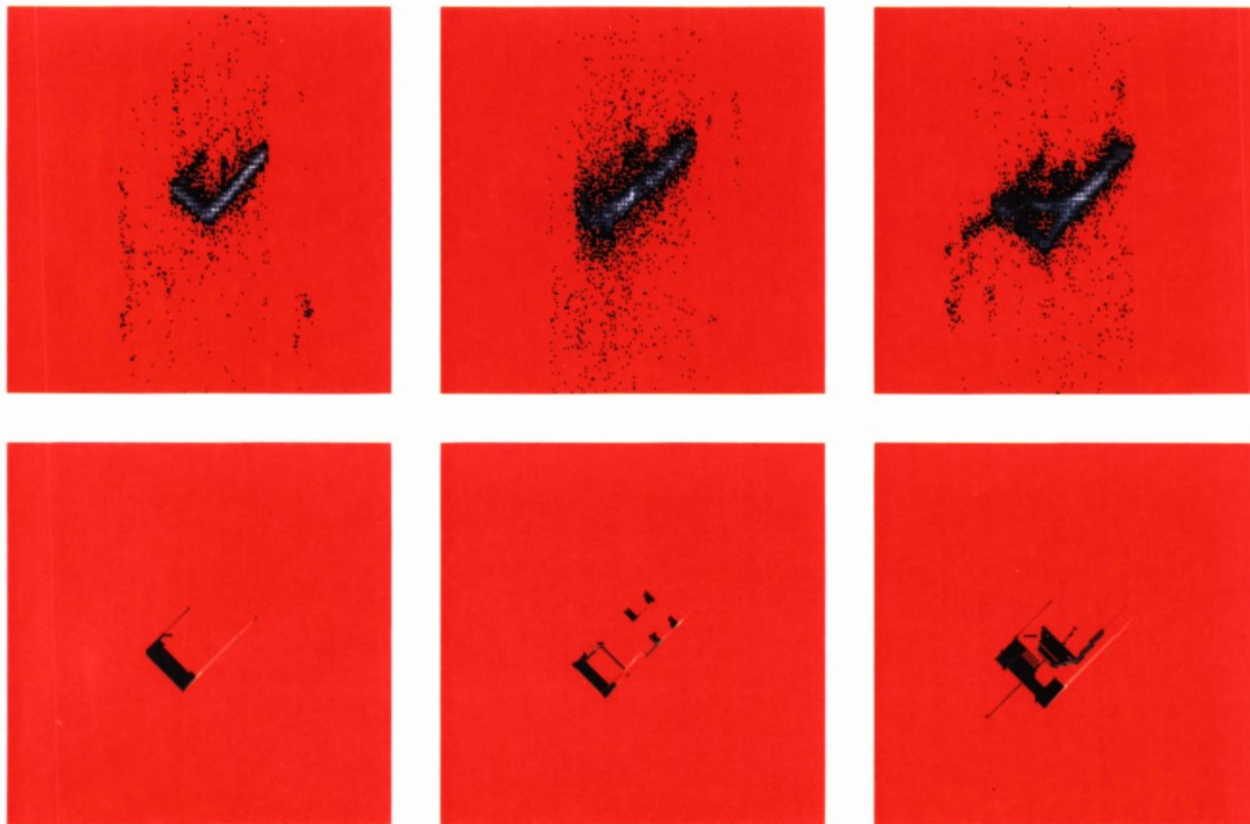


Figure 13. Comparison of overhead view "count images" for three real targets and three synthetic targets at nearly identical azimuth angles near 45° . In all cases, target range is 1000 m and reprojection pixel sizes are 0.10 m. Top row shows views corresponding to TSLR M113A2 APC, M35 truck, and M60A1 tank (the respective ground-truth angles are 42° , 44° , and 41°). Bottom row shows corresponding views for SYNTH M113, M35, and M60 (all ground-truth angles are exactly 45°).

4.4 Hough Transform (HT) and Peak Detection

The results of applying the HT algorithm to the “count images” shown in Figures 11 and 12 are shown in Figures 14 and 15, respectively. A horizontal width of 180 pixels was selected for the HT image in order to provide an angle bin of 1° . Therefore, the first angle bin of the HT corresponds to -90° and the last angle bin to $+89^\circ$. (The middle bin corresponds to 0° .)

Figure 16 depicts the authors’ definition of the HT angle (the direction the front of the target is facing is irrelevant; only the orientation of the target’s main axis matters). Figure 17 further illustrates the relationship between the azimuth angle and the HT angle. Part of the difficulty in relating both angles arises from the fact that the azimuth angle is defined in the range from 0° to 360° (excluded) and the HT angle is defined in the range from -90° to $+90^\circ$ (excluded). It is only for azimuth angles in the range from 0° to 90° that the azimuth angles and the HT angles have (ideally) the same numerical values.

Returning to Figures 14 and 15, observe that the main peak of the HT generally corresponds to the correct azimuth for the target (plus or minus 180° because of the ambiguity of the HT). By playing movies of the HT images, one can see the peak of the HT move across the HT plane as the target rotates on its hypothetical turntable. Further examples are given on Figures 18 and 19. Even though both figures correspond to near-tail-on views, observe that the appearances are quite different because the side that is visible is different in both figures.

As expected, the target orientation estimate provided by the HT is generally off by about 90° in the case of front views and rear views of targets. In effect, the front or rear edge of the target is mistaken for one of the sides of the target. It is a fact that the HT will *never* be able to reliably distinguish between front views, rear views, and side views of objects. The solution to this problem is never to trust any HT estimate that is near -90° or $+90^\circ$ and to always use the FTs corresponding to front and rear views when suspicious numbers are produced by the HT. Therefore, if the angle returned by the HT is in the suspicious range, the FTs will be used over four angular sectors that are centered over the four cardinal azimuth angles (instead of over just the standard pair of diametrically opposed sectors).

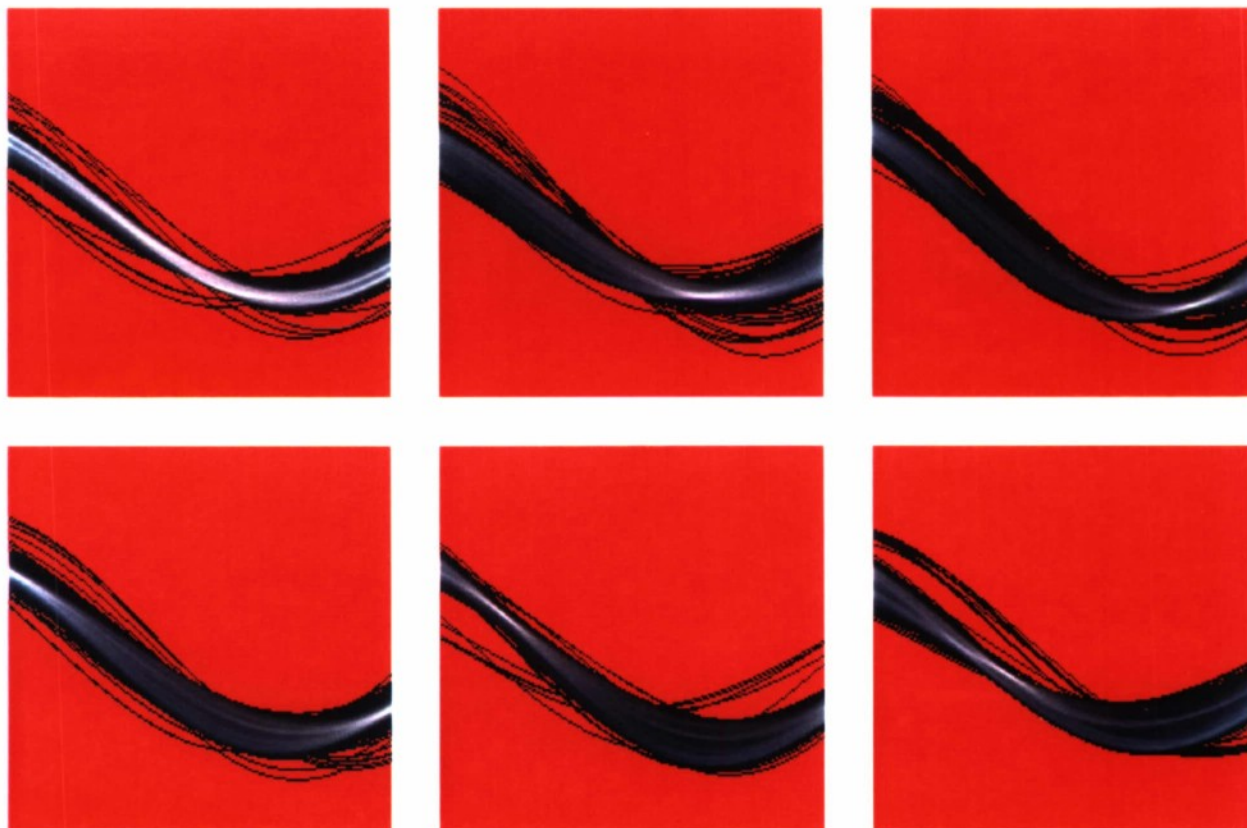


Figure 14. HTs of the local “count images” for the TSLR M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.

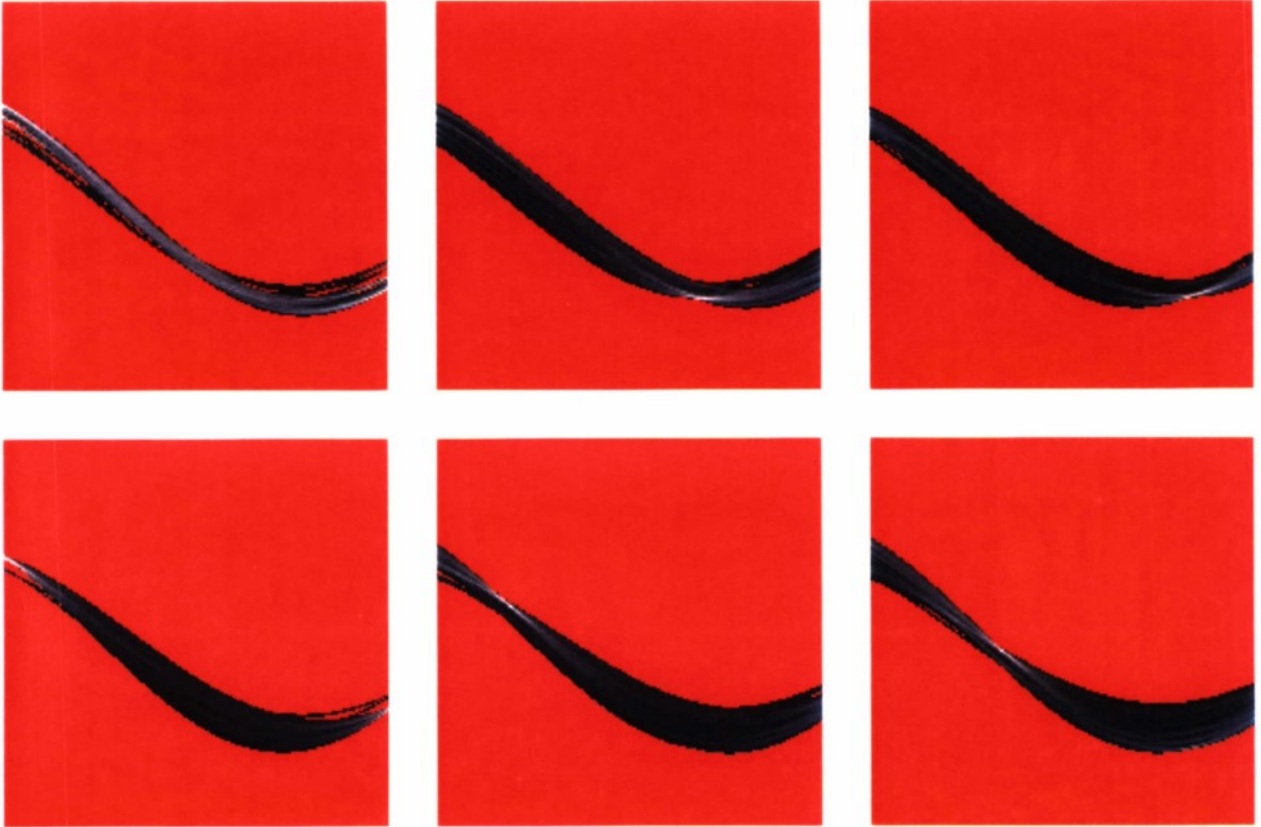


Figure 15. HTs of the local “count images” for the SYNTH M35 truck at 1000 m with resolutions of 0.05 mrad and reprojection pixel sizes of 0.10 m.

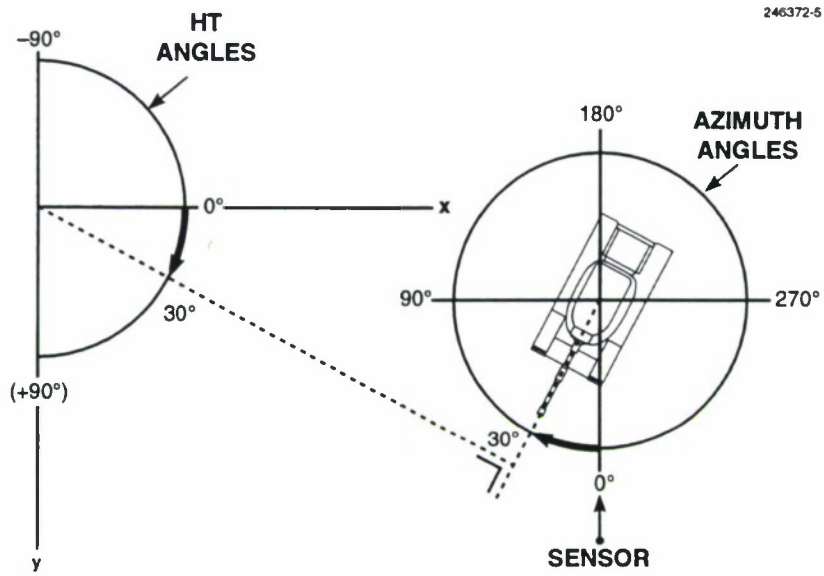


Figure 16. HTs angle convention: case of a tank with a 30° azimuth angle.

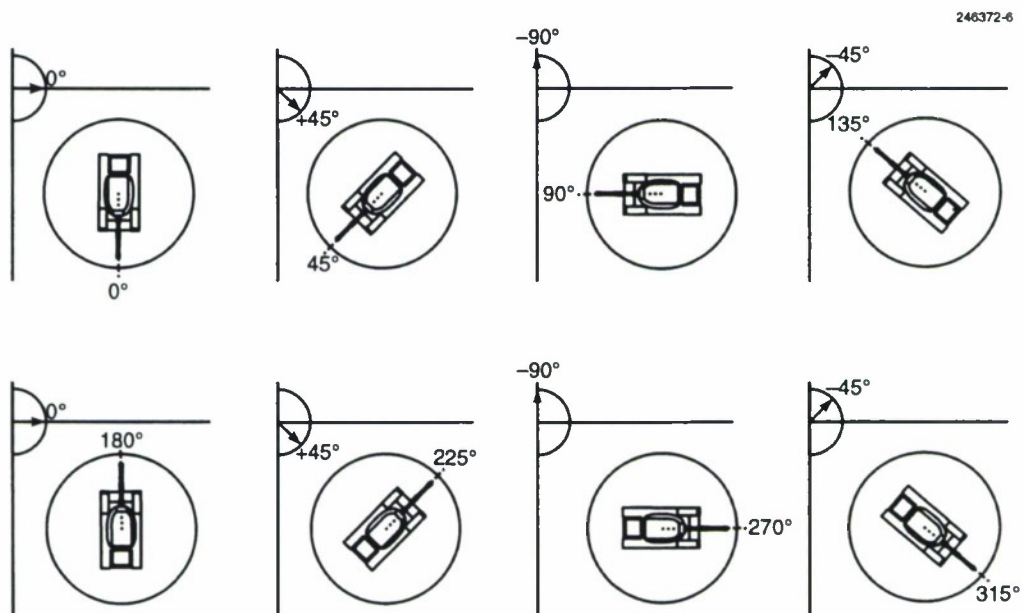


Figure 17. Relation between azimuth angle and HTs angle for a variety of situations.

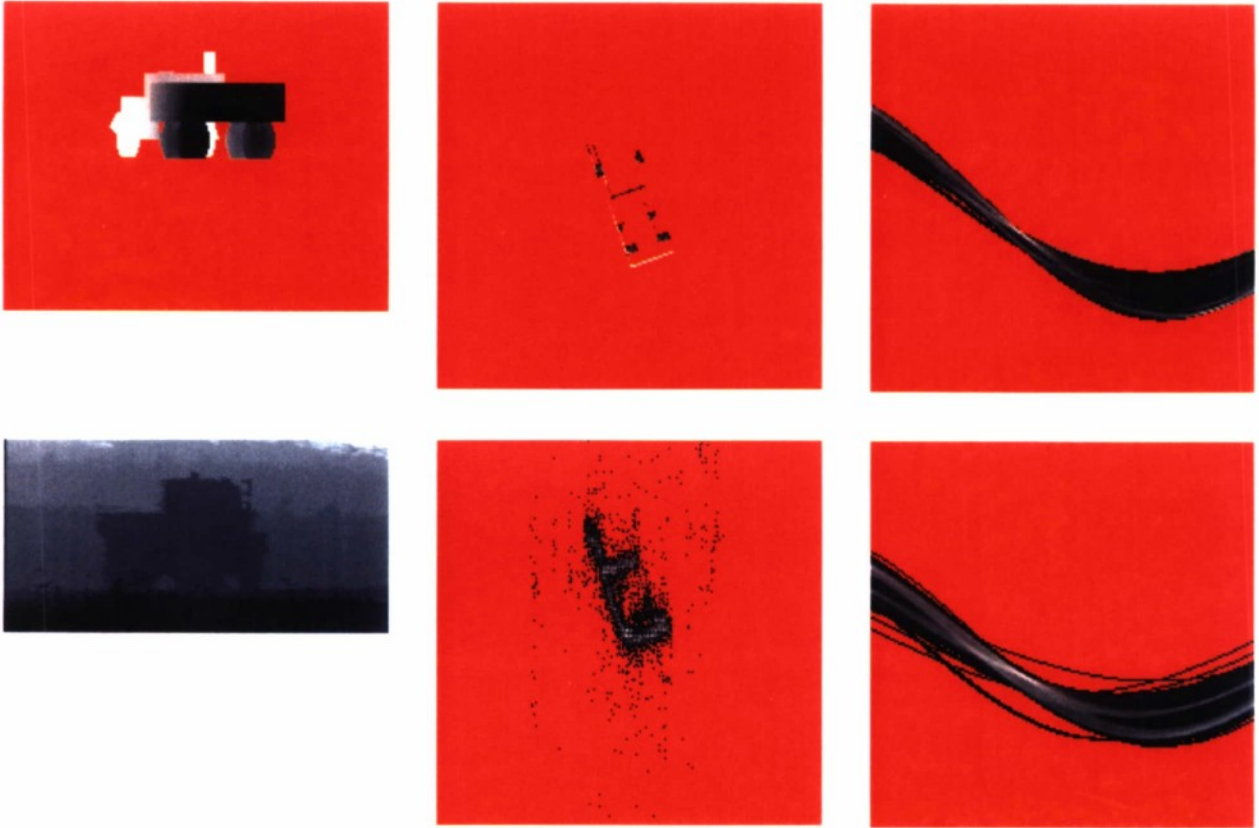


Figure 18. Raw range images, overhead view "count images", and HTs for SYNTH (top row) and TSLR (bottom row) M35 truck. Ground-truth azimuth for SYNTH case is 160° , and HT angle estimate is $162^\circ (\pm 180^\circ)$. Corresponding numbers for TSLR case are 161° and 161° .

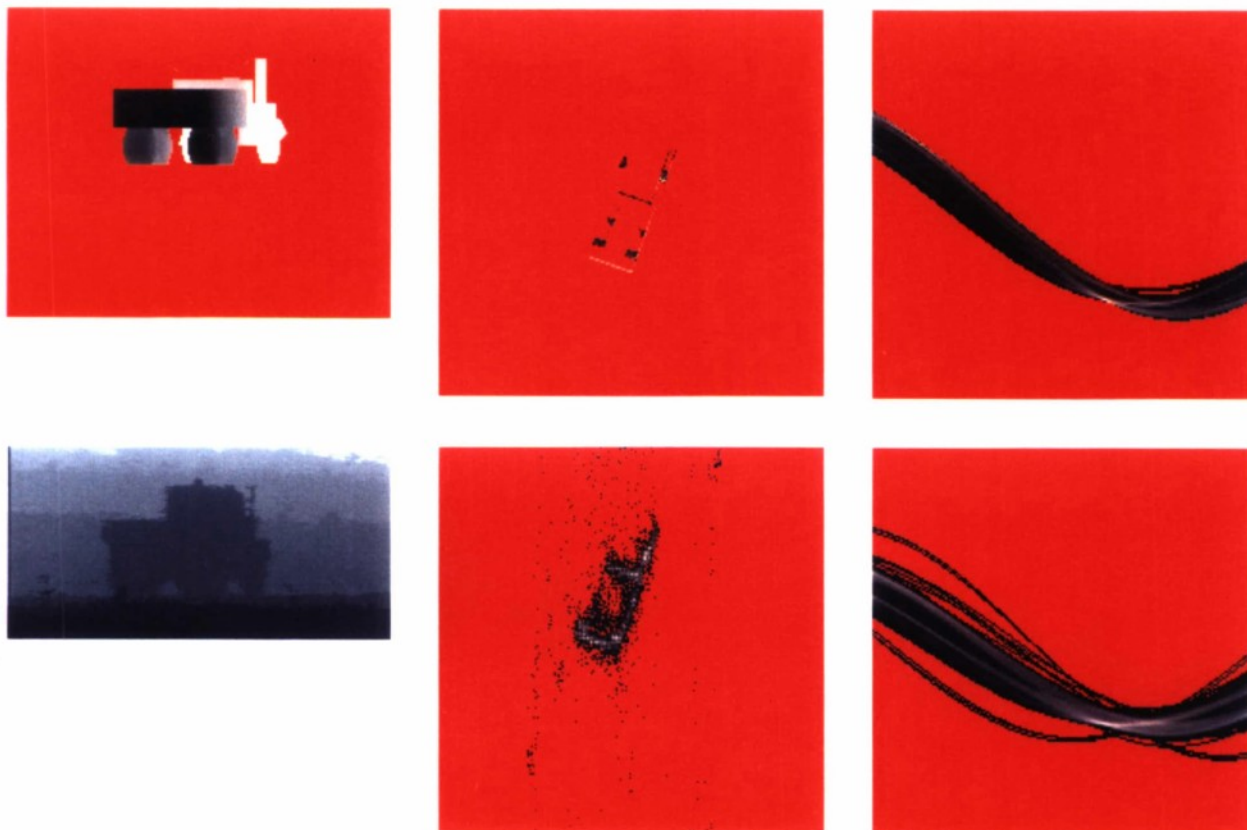


Figure 19. Raw range images, overhead view “count images”, and HTs for SYNTH (top row) and TSLR (bottom row) M35 truck. Ground-truth azimuth for SYNTH case is 200° , and HT angle estimate is $199^\circ (\pm 180^\circ)$. Corresponding numbers for TSLR case are 198° and 197° .

To extract the target angle estimate from the HT image, one uses the previously described peak extractor. To quantify the performance of the combined HT and peak extractor, these algorithms were applied to the three TSLR data sets corresponding to the M113A2 APC (64 images), M35 truck (70 images), and M60A1 tank (52 images), all at 1000 m and 0.05-mrad resolution. As previously explained in Section 2, one had to establish the ground-truth azimuth angle for each of the 186 images involved. In addition, the performance of the algorithms on the synthetic equivalents of the TSLR data sets were also quantified. Each of the SYNTH data sets has a total of 72 images. Of course, the synthetic data sets have the advantage that the ground truth is immediately available for all 216 images. The performance evaluation for all 402 images of the six data sets (TSLR and SYNTH) is presented in the form of graphs that show the signed error of the HT angle estimate (with allowance for the possible 180° ambiguity) as a function of the ground-truth azimuth angle. (Note that, in the case of the TSLR data, the horizontal spacing between successive data points is not constant because of errors in the actual angular positioning of the target in the field.)

The graphs of Figure 20 correspond to the TSLR sequences. The upper-left graph corresponds to the TSLR sequence M113A2-1000-C6 and is shown with a vertical error scale covering the range of $[-10^\circ, +10^\circ]$. Data points that appear to be missing either do not exist or are outside the error range (as is often the case for front and rear views). All remaining graphs have an error scale of $[-90^\circ, +90^\circ]$ that covers all possible errors. In this presentation, the near- 90° errors for the near-front views (around 0° azimuth) and the near-rear views (around 180° azimuth) are quite visible. The three graphs with the full error scale correspond to the three TSLR sequences M113A2-1000-C6, M35-1000-C6, and M60A1-1000-C6.

Figure 21 is the equivalent of Figure 20 for the three SYNTH sequences M113-1000-005-T, M35-1000-005-T, and M60-1000-005-T.

Finally, Figure 22 corresponds to the case where the “1D HT” discussed in Section 3 is applied to the SYNTH data sets (instead of the customary 2D HT). Note that the extents of the angular sectors where the near- 90° errors usually occur have been considerably reduced. However, new errors appear near broadside. In spite of this, there might be a way to use the 1D and 2D HTs in combination to get excellent orientation estimates everywhere except for views that correspond exactly to front or rear views of targets. However, the 1D HT does not appear to work reliably on real data.

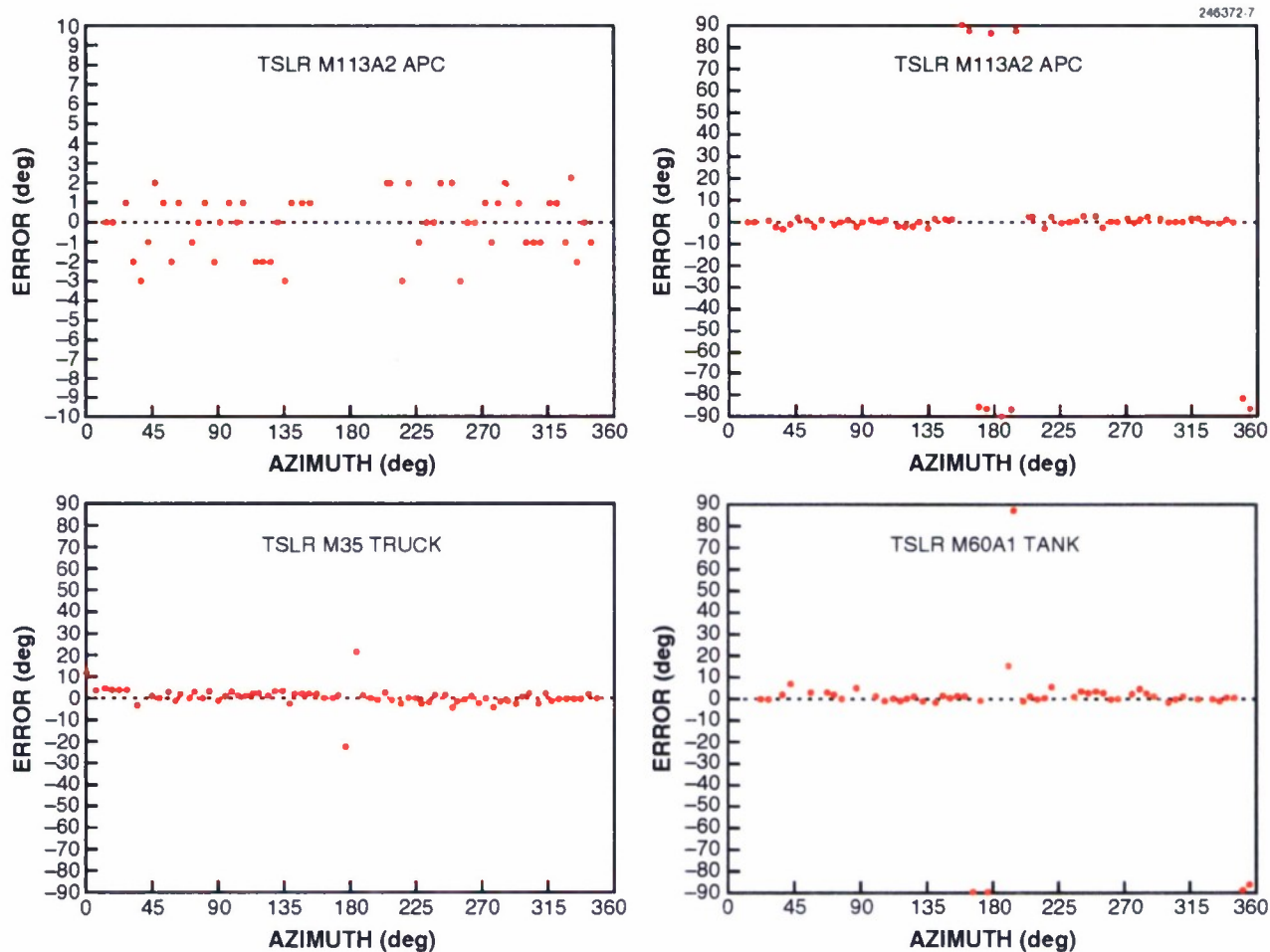


Figure 20. Signed error of HT azimuth estimate for the TSLR M113A2 APC, M35 truck, and M60A1 tank. While the upper-left graph has a scale limited to an absolute error of 10° , the other three graphs show the full error range.

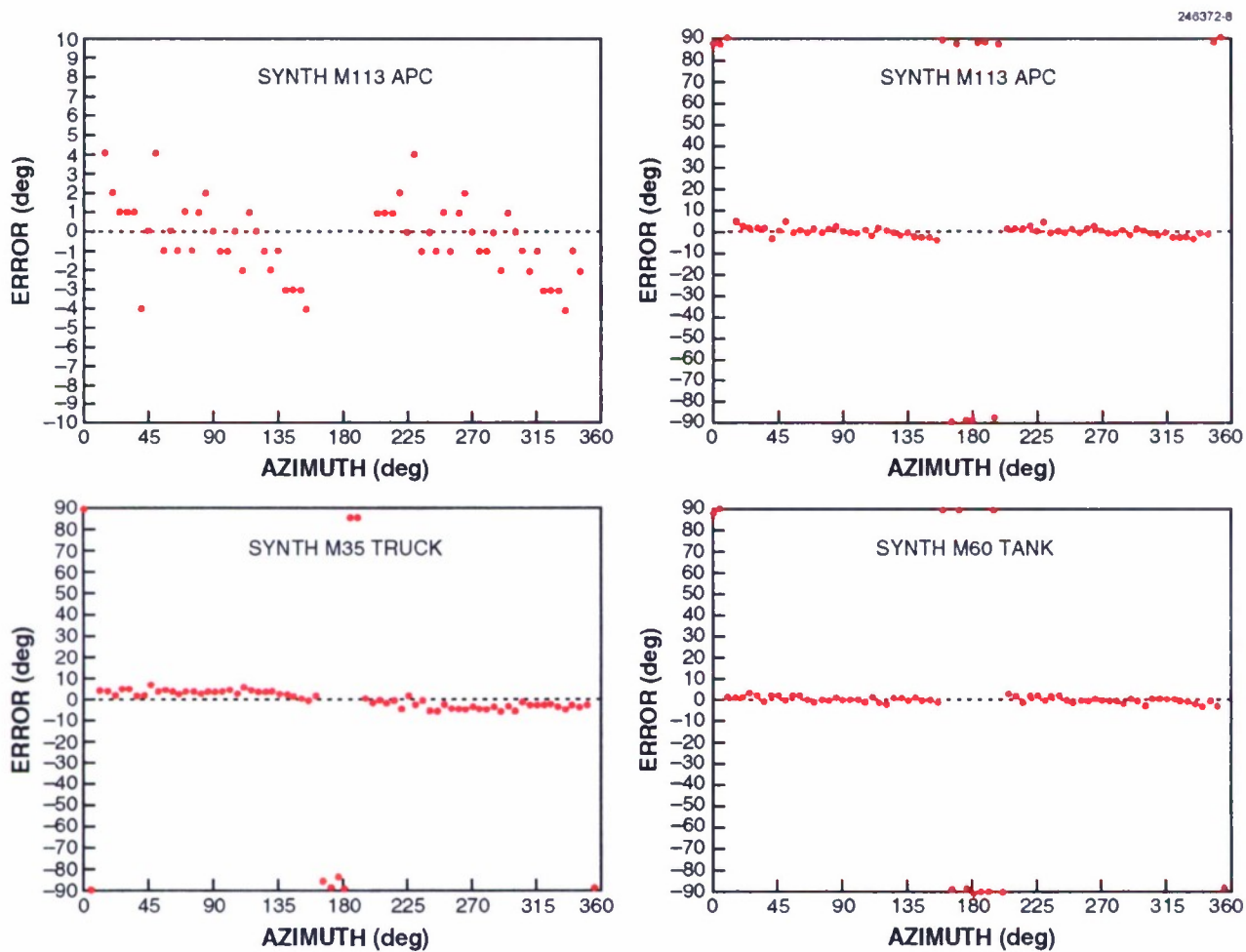


Figure 21. Signed error of HT azimuth estimate for the SYNTH M113 APC, M35 truck, and M60 tank. While the upper-left graph has a scale limited to an absolute error of 10° , the other three graphs show the full error range.

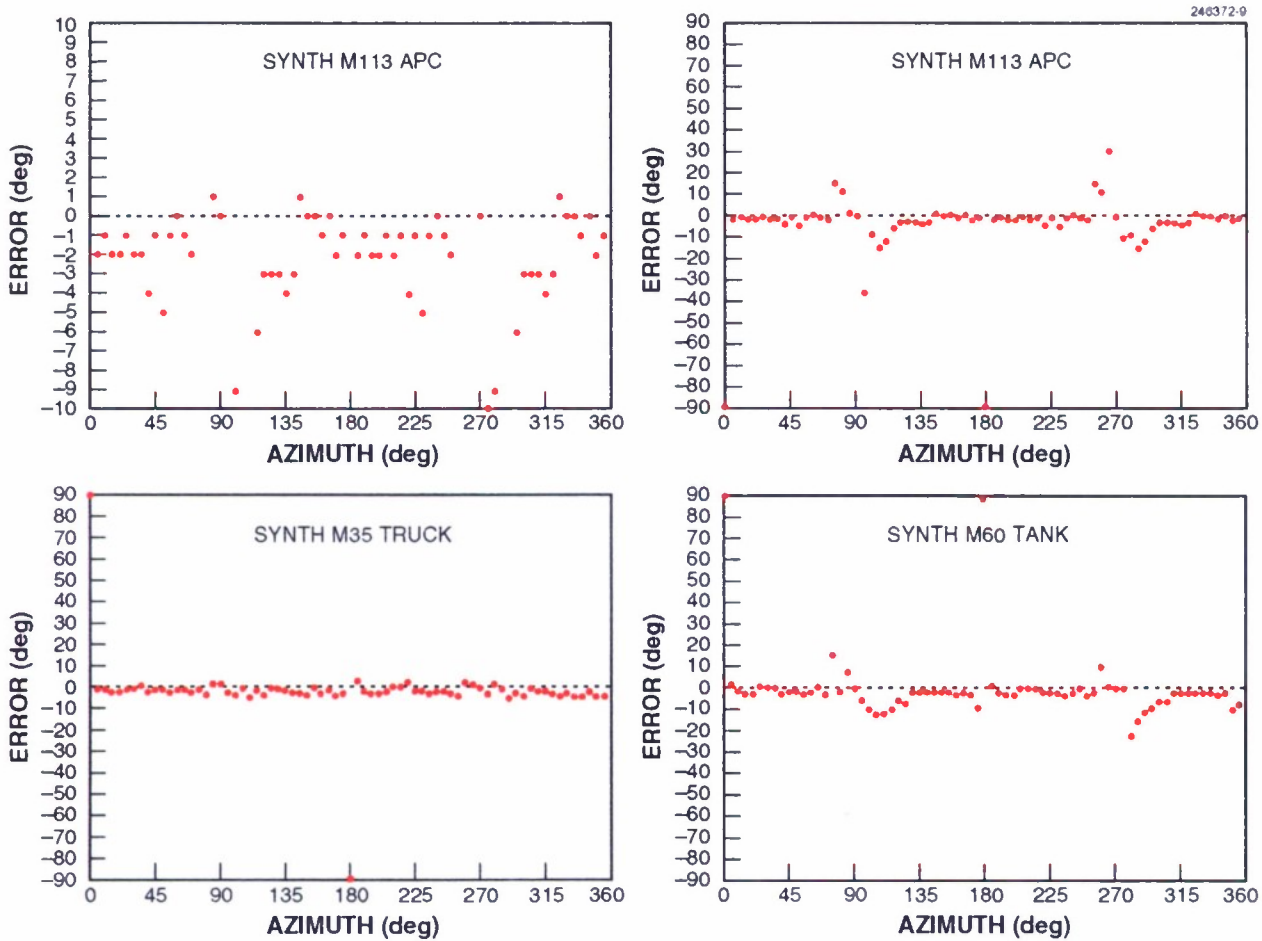


Figure 22. Signed error of 1-D HT azimuth estimate for the SYNTH M113 APC, M35 truck, and M60 tank. While the upper-left graph has a scale limited to an absolute error of 10° , the other three graphs show the full error range.

5. CONCLUSION

This report gives the status of the authors' initial work in support of the UGV Demo II Program in the RSTA area, where the authors' task is to design the ATR system for the RSTA ladar.

Because this ladar has not yet been built, a major concern was the acquisition of appropriate data for beginning to develop and test the recognition system. A review was presented of the organization of the available TSLR data into useful sequences, each corresponding to successive views of a target at increasing viewing angles (as if the target was rotating on a hypothetical turntable). One limitation of that data base is that most of the targets are all at about 1000 m. Even though this is the nominal range for recognizing targets with the RSTA ladar, it would be useful to have similar sequences at a variety of ranges around 1000 m. Another limitation is that the angular resolutions of the available TSLR data are not representative of that of the planned RSTA ladar. The generation and uses of synthetic data were also discussed (both for augmenting the development and test data base and for building FT models). Finally, the HBS data was briefly described.

A major part of the work to date has focused on the recognition part of the end-to-end system. The main idea is to reproject forward-looking ladar data into overhead views of scenes and to apply sets of azimuth-dependent FTs to recognize targets. So far, the target and the ladar have both been assumed to lie on more or less horizontal terrain. Thus, the reprojections are performed vertically. Under less ideal conditions, one would want to reproject the images orthogonally to the local terrain in order to prevent distortions in the overhead views of targets. Clearly, this will necessitate detailed knowledge of the terrain near the target as well as detailed position and attitude information for the sensor. If the target is unobscured, it seems feasible that the ground orientation could be found by image processing techniques exploiting the availability of range information in the vicinity of the target. This approach will fail if the target is obscured from below in such a way that the terrain near the target is masked (this would be the case for a target behind a stone wall or in a ditch). In this case, the only solution may be to use map and GPS information. Of course, larger sets of FTs could be built to account for a variety of target attitudes, but the construction of large numbers of templates may be unfeasible, especially if the design of the templates remains a manual task. So far, the authors' work did not fully address the issue of articulation of target parts. However, some of the techniques that were successfully used in connection with downlooking ladar imagery [4] could be applied here. Indeed, separate sets of templates could be used for each of the articulated parts. For example, a first set of templates could be used to look for the hull of a tank. Then, when the most probable position and orientation of the hull is found, a set of templates would be spun in the vicinity of the articulation point of the turret on the candidate hull. (An example of a tank with a rotated turret is visible in the TSLR image of the tank in Figure 13.)

Other initial contributions have been the development of the HLV interest image for target detection and of the HT (and associated peak detector) for finding the orientation of the target's long axis as a way of cueing the search with the FTs.

Many other relevant algorithms have been previously implemented under the SKETCH system [19], but they will need to be recoded to run under Khoros [7] and to be modified to satisfy the needs of the RSTA radar.

.

REFERENCES

1. E.G. Mettala, "The OSD tactical unmanned ground vehicle program," in *Proceedings of Image Understanding Workshop*, pp. 159-171, San Diego, CA, 26-29 January 1992.
2. B. Gothard, R.L. Spomer, J. Medina, and W. Chun, "Design issues with the SSV platform," in *Mobile Robots VII, Proceedings of SPIE, Vol. 1831*, Boston, MA, 18-20 November 1992, pp. 320-331.
3. J.G. Verly, R.L. Delanoy, and D.E. Dudgeon, "Machine intelligence technology for automatic target recognition," *Lincoln Laboratory Journal*, 2(2), pp. 277-311 (1989).
4. R.L. Delanoy, J.G. Verly, and D.E. Dudgeon, "Machine intelligent automatic recognition of critical mobile targets in laser radar imagery," *Lincoln Laboratory Journal*, 6(1), pp. 161-186 (1993).
5. J.G. Verly, R.L. Delanoy, and C.H. Lazott, "Principles and evaluation of an automatic target recognition system for synthetic aperture radar imagery based on the use of functional templates," in *Automatic Target Recognition III, Proceedings of SPIE, Vol. 1960*, Orlando, FL, 14-16 April 1993, pp. 57-71.
6. J.E. Nettleton and S. Holder, "Active/passive laser radar field test (Fort A.P. Hill, VA, Drop Zone)," Technical Report AMSEL-NV-TR-0091, Center for Night Vision and Electro-Optics, Fort Belvoir, VA (1990).
7. K. Konstantinides and J.R. Rasure, "The Khoros software development environment for image and signal processing," in *IEEE Transactions on Image Processing*, 3(3) (1994).
8. J.E. Bevington, "Laser radar ATR algorithms (phase III final report)," Alliant Technical Report Techsystems, Inc., Hopkins, MN (1992).
9. J.R. Beveridge, S. Hennessy, D.P. Panda, B. Hoff, and T. Yachik, "November 1993 Fort Carson RSTA data collection - final report," Technical Report (1994).
10. R.L. Delanoy, J.G. Verly, and D.E. Dudgeon, "Pixel-level fusion using interest images," MIT Lincoln Laboratory, Lexington, MA, Technical Report TR-979 (1993).
11. R.L. Delanoy, J.G. Verly, and B.D. Williams, Private communication, 1992.
12. J.G. Verly and R.L. Delanoy, "Adaptive mathematical morphology for range imagery," *IEEE Transactions on Image Processing*, 2(2), pp. 272-275 (1993). (Also see 2(4), pp. 550-551, 1993.)

REFERENCES

(Continued)

13. J.G. Verly and R.L. Delanoy, "Some principles and applications of adaptive mathematical morphology for range imagery," *Optical Engineering*, 32(12), pp. 3295-3306 (1993).
14. R.L. Delanoy, J.G. Verly, and D.E. Dudgeon, "Functional templates and their applications to 3D object recognition," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, San Francisco, CA, 23-26 April 1992, pp. III-141-III-144.
15. R.L. Delanoy and S.W. Troxel, "Machine intelligent gust front detection," *Lincoln Laboratory Journal*, 6(1), pp. 187-212 (1993).
16. R.L. Delanoy and J.G. Verly, "Computer apparatus and method for fuzzy template shape matching using a scoring function," *United States Patent*, No. 5,222,155, 22 June 1993.
17. J.G. Verly, B.D. Williams, and R.L. Delanoy, Private communication, 1991.
18. T-J. Fan, *Describing and Recognizing 3-D Objects using Surface Properties*, New York, NY: Springer-Verlag (1990).
19. R.L. Walton, J.G. Verly, and P.L. Van Hove, "Sketch 4b: An Image Understanding Operating System," MIT Lincoln Laboratory, Lexington, MA, Lincoln Manual LM-163 (1989).

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188), Washington, DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE 21 March 1995	3. REPORT TYPE AND DATES COVERED Technical Report	
4. TITLE AND SUBTITLE Progress Report on the Development of the Automatic Target Recognition System for the UGV/RSTA LADAR			5. FUNDING NUMBERS C — F19628-95-C-0002 PR — 322 PE — 61101E	
6. AUTHOR(S) Jacques G. Verly, Dan E. Dudgeon, and Richard T. Lacoss				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Lincoln Laboratory, MIT P.O. Box 73 Lexington, MA 02173-9108			8. PERFORMING ORGANIZATION REPORT NUMBER TR-1006	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Advanced Research Projects Agency/ISTO ARPA/CEXEC 370 No. Fairfax Dr. Arlington, VA 22203-1714			10. SPONSORING/MONITORING AGENCY REPORT NUMBER ESC-TR-94-111	
11. SUPPLEMENTARY NOTES None				
12a. DISTRIBUTION/AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
13. ABSTRACT (Maximum 200 words) This report describes the authors' initial work under ARPA Unmanned Ground Vehicle (UGV) Demo II Program in the reconnaissance, surveillance, and target acquisition (RSTA) area. The task is to develop the automatic target recognition (ATR) system that will process the imagery from the RSTA laser radar (ladar). A real-time demonstration of this capability is scheduled for the "Demo II" of 1996. All major components of this end-to-end ATR system are discussed, and more details are provided for critical elements that have been built and tested so far. A major topic of interest is the use of sets of azimuth-dependent functional templates and functional template correlation forward-looking ladar data. Other topics are the use of height-limited verticality as an interest image for focusing attention and of the Hough transform for getting a preliminary estimate of target orientation. Because the RSTA ladar remains to be procured and built, the report describes also the alternate sources of data that are being used to develop and test the system elements built to date.				
14. SUBJECT TERMS Ladar Ghost artifact Hough transform Tri-service laser radar Limited-height verticality Functional templates Range imagery Vertical reprojection			15. NUMBER OF PAGES 94	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT Same as Report	